Automatic logarithm and associated measures

Rostislav Grigorchuk * Roman Kogan Yaroslav Vorobets

November 28, 2018

Abstract

We introduce the notion of the *automatic logarithm* $\mathcal{L}_{\mathcal{A},\mathcal{B}}$ with the purpose of studying the expanding properties of Schreier graphs of action of the group generated by two finite initial Mealy automata $\mathcal{A}$ and $\mathcal{B}$ on the levels of a regular d-ary rooted tree $\mathcal{T}$, where $\mathcal{A}$ is level-transitive and of bounded activity. $\mathcal{L}_{\mathcal{A},\mathcal{B}}$ computes the lengths of chords in this family of graphs. Formally, $\mathcal{L}$ is a map $\partial\mathcal{T} \to \mathbb{Z}_d$ from the boundary of the tree to the integer p-adics whose values are determined by a Moore machine. The distribution of its outputs yields a probabilistic measure μ on $\partial\mathcal{T}$, which in some cases can be computed by a Mealy-type machine (we then say that μ is *finite-state*). We provide a criterion to determine whether μ is finite-state. A number of examples illustrating the different cases with $\mathcal{A}$ being the adding machine is provided.

1 Introduction

The maps and the measures considered in this paper arise from the study of properties of Schreier graphs associated with automaton semigroups and groups acting on words over a finite alphabet and regular rooted trees.

The problem of studying the distribution of lengths of chords (to be defined below) in the graph of action of two initial automata gives rise to the *automatic logarithm*, a map defined by an automaton that outputs these lengths. The distribution of the lengths of chords is then seen as the image of the uniform Bernoulli measure by the action of the automatic logarithm. When the automatic logarithm is invertible, the distribution is uniform. Otherwise, the resulting distribution is an interesting object of study. In certain cases, such distributions only have a finite number of restrictions

*The author was supported by NSA grant H98230-15-1-0328 and by Simons Foundation Collaboration Grant for Mathematicians, Award Number 527814.

to cylinders (we call them **finite-state**), and we provide a sufficient condition for this to happen, as well as examples when it does not.

Given a finite initial Mealy (or Moore) type automaton $\mathcal{A}_q$ over a finite alphabet X, one can define a map $\hat{\mathcal{A}}_q$ on the space of sequences (words) over the alphabet X. Maps of this type usually have a very complicated dynamical nature and may transform relatively simple measures on the space $X^{\mathbb{N}}$, like for instance Bernoulli or Markov measure, into complicated ones. The study of such measures were initiated in [**AKP**], [**Ryab**] and [**Krav**].

Given a family of finite automata $\mathcal{A}_q, \mathcal{B}_s, \dots, \mathcal{C}_t$, using the operation of composition of automata one can generate a semigroup $\mathcal{S} = \langle \mathcal{A}_q, \mathcal{B}_s, \dots, \mathcal{C}_t \rangle_{\text{sem}}$ or even a group $\mathcal{G} = \langle \mathcal{A}_q, \mathcal{B}_s, \dots, \mathcal{C}_t \rangle_{\text{gr}}$ if the automata are invertible. A particularly interesting case is when the group $\mathcal{G}$ is generated by a family which comes from the one non-initial invertible automaton $\mathcal{A}$ by using all its states for generating. Such groups are called automaton groups (or self-similar groups) and play an important role in group theory and areas of its applications [**GB**], [**Nek**], [**GNS**], [**GNSunic**]. They naturally act by automorphisms on a d-regular rooted tree $\mathcal{T}$ (d is the cardinality of the alphabet X) and on its boundary ∂T. These actions are induced by the corresponding actions on the set of finite (and, respectively, infinite) words over the alphabet X. The operation of composition of automata corresponds to the composition of the associated maps.

Another direction of development is study of the Schreier graphs (also called orbital graphs) given by the action of a group on levels of the tree or on its boundary (i.e. on finite or infinite words). These graphs have self-similarity features and give a good approximation to many important fractal sets including the Julia sets of the rational mappings of $\mathbb{C}$. There are examples of the automata given by a small number of states that are believed to produce families of expander graphs (two of them are considered in this article). No rigorous proof of this is known, but there are results showing that at least these families are the so called asymptotic expanders [**GrigExp**], and that the growth of their diameters is slow [**PakMal**] (as should be in the case of expanders).

Among automorphisms of the rooted trees, the most famous is the *adding machine* automorphism defined by the automaton shown in Figure 2a, which we denote $\mathcal{O}$. The *portrait* of this automorphism is shown in Figure 4. It acts on finite strings of symbols as the addition of 1 when the strings are interepreted as the natural numbers in the d-adic expansion (the diagram in Figure 2a is for the case of the binary alphabet). The group $\mathcal{G}(\mathcal{O})$ is an infinite cyclic group (one of the states of $\mathcal{O}$ corresponds to the identity map). If o is the nontrivial state of $\mathcal{O}$, and if another initial automaton $\mathcal{A}_a$ is given, then one can consider the semigroup $\langle A_a, O_o \rangle_{\text{sem}}$ (or a group $\langle \mathcal{A}_a, \mathcal{O}_o \rangle_{\mathrm{gr}}$ if $\mathcal{A}_a$ is invertible), and study its properties as well as the sequence $\{\Gamma_n\}_{n=1}^{\infty}$ of graphs of action on the levels $n = 1, 2, \ldots, n \ldots$ of the tree (Figure 3 gives an impression of how the graph Γ_n may look). The questions about combinatorial and spectral properties of graphs $\{\Gamma_n\}$ is the subject of many investigations [**GBspec**, **GZ**, **GSHanoi**], and in particular, the question about the growth of the diameters of $\{\Gamma_n\}$ and about the expansion properties of the family $\{\Gamma_n\}_{n=1}^{\infty}$ are among the central.

In this paper we focus on study of the dynamical and combinatorial properties of the pair $(\mathcal{A}_a, \mathcal{O}_o)$. This unexpectedly leads us to the notion of the "logarithm" of $\mathcal{A}_a$ with respect to the adding machine $\mathcal{O}_o$, which we denote $\mathcal{L}$, and which is a specially defined map on the set of finite and infinite sequences with vales in d-adic numbers, or words over X that represent them. Then we show that in the case of the binary alphabet, the logarithm map can be define by a finite state automaton (Theorem 6.3) and provide the construction for it. We then analyze the distribution of the lengths of the "chords" (again we appeal to Figure 3 which gives an impression of what we mean by the chord). This leads us to the considerations started in the [**Krav**, **GKV1**] about the nature of the image of the Bernoulli (or, more generally, Markov) measure under the automaton map, in the case the map is given by the "logarithm" automaton $\mathcal{L}$. The distribution of the chords is given by the image $\mu = \mathcal{L}_*(\nu)$ of the uniform Bernoulli measure ν on $X^{\mathbb{N}}$, which in some important cases (for instance, given in the Example 8.5 and Theorem 8.6) is a Markov measure, but in some other interesting cases (like the Example 8.7) is a more complicated type of measure.

To study μ, we introduce the notion of the automaton associated with a measure, the notion of a finite state measure, of a self-similar measure, and show that Markov measure is a finite state measure, but not every finite state measure is Markov. Multiple examples illustrating the definitions and results are provided (we mark the ends of examples with $\triangle$ instead of the traditional $\square$).

2 Preliminaries

2.1 Endomorphisms of rooted trees

By a finite **alphabet** X of size d, we mean a finite set of cardinality d. We will usually use the following alphabets: the sets $\{0, 1, \ldots, d-1\}$, and finite subsets $X \subset [0, 1]$ (in the latter case, the values of the real numbers constituting X will be important in addition to its cardinality).

For a word w in X, $|w|$ denotes its length, and w_i denotes the i'th character for $0 \le i \le |w| - 1$. The numbering of characters starts from 0, so $w = w_0 w_1 \ldots w_{|w|-1}$.

If v is another word (or a character), wv is the concatenation of the two.

X^* denotes all finite words over X:

$$X^* := \{a_0 \dots a_{n-1} \ : \ a_i \in X, n \in \mathbb{N} \cup \{0\}\}$$

Let $\mathcal{T}$ be a rooted graph with the vertex set $V = X^*$, edge set $F = \{(w, wa) \ : \ w \in X^*, a \in X\}$, and the root being the empty word. This graph is a d-**regular rooted tree**.

The n'th level X^n of the tree $\mathcal{T}$ is the set of words of length n.

An **endomorphism** of the rooted tree $\mathcal{T}$ is a map g from X^* to itself that preserves the levels and maps adjacent vertices to adjacent vertices. An **automorphism** is an invertible endomorphism.

The **boundary** of the tree $\mathcal{T}$ is the set $X^{\mathbb{N}}$ of infinite sequences in X:

$$\partial\mathcal{T} := \{a_0 a_1 a_2 \dots \ : \ a_i \in X, i \in \mathbb{N}\}$$

∂T is supplied with the Tychonoff product topology that makes it homeomorphic to a Cantor set. Geometrically, the boundary can be viewed as a set of geodesic paths starting at the root and going to infinity.

Let σ_r denote the operation that deletes the last character of a word: for $w \in X^*$ and $a \in X$, $\sigma_r(wa) := w$. Then ∂T can be obtained as the inverse limit of the directed system of levels $\{X^n \ : \ n \in \mathbb{N}\}$ with the projections $\psi_{m,n} : X^n \to X^m$ given by $\psi_{m,n} := \sigma_r^{n-m}$ (i.e., discarding the last $n - m$ characters).

2.2 Mealy and Moore machines

Definition 2.1 *A* ***Mealy machine****, or a* ***finite initial automaton with output****, is a hextuple $\mathcal{A}_q = (S, q, X, Y, \pi, \lambda)$, where*

- *X is a (finite) input alphabet;*
- *Y is a (finite) output alphabet;*
- *S is a (finite) set of states;*
- *$q \in S$ is the initial state;*
- *$\pi : S \times X \to S$ is the transition map*
- *$\lambda : S \times X \to Y$ is the output map*

When the initial state of the automaton is understood from the context, we drop the subscript and write $\mathcal{A}$ instead of $\mathcal{A}_q$ to denote it.

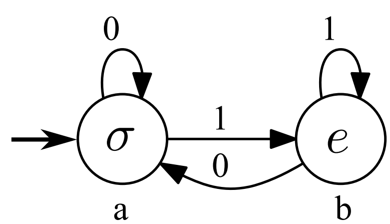
We write π_s, λ_s for restrictions of these functions to the state s, defining $\pi_s(x) := \pi(s,x)$ and $\lambda_s(x) := \lambda(s,x)$.

The functions π and λ also act on words in the alphabet X via the following recursive relations (for $x \in X$, $w \in X^*$):

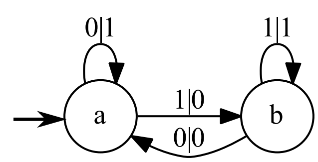
$$\pi(s,xw) := \pi(\pi(s,x),w);$$
$$\lambda(s,xw) := \lambda(s,x)\lambda(\pi(s,x),w).$$

In the same way, π_s and λ_s, for $s \in S$, act on words $w \in X^*$. Additionally, we may write $\pi(w)$ for $\pi_q(w)$ (and similarly, $\lambda(w)$ for $\lambda_q(w)$), when the initial state q is understood from the context.

The **diagram** of an automaton $\mathcal{A}_q$ is a labeled graph with the vertex set S, edge set $E = \{(s,\pi(s,x)) \ : \ s \in S, x \in X\}$, with label $x : \lambda(s,x)$ on the edge $(s,\pi(s,x))$. The initial state q is marked with a special arrow (which doesn't start at a state). An example of such diagram for the Lamplighter automaton is shown in Figure 1b.



(a) Diagram with element of the symmetric group on vertices



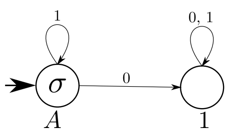
(b) Diagram with output marked on edges

Figure 1: Two ways to draw the Lamplighter automaton

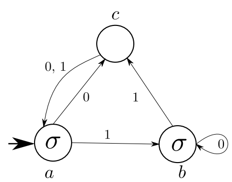
An automaton $\mathcal{A}$ is **invertible** if λ_s is invertible for all $s \in S$ (that is, if $\lambda_s \in S(X)$, where $S(X)$ is the symmetric group on X). The endomorphism g given by an invertible automaton $\mathcal{A}$ is invertible, and the automaton for g^{-1} (which we denote as $\mathcal{A}^{-1}$) can be constructed from the diagram of $\mathcal{A}$ by flipping the input and output on the edges.

In the case when an automaton is invertible, we can draw the **diagram** of the automaton without specifying its output on the arrows. Instead, the state s is marked by the element of the symmetric group $\lambda_s \in S(X)$. If λ_s is the trivial permutation, we call the state s **passive**, and call it **active** otherwise.

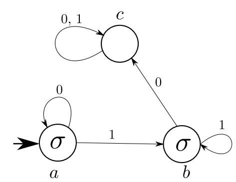
When $X = \{0, 1\}$, we write σ for the nontrivial permutation of X (i.e. $\sigma(0) = 1, \sigma(1) = 0$). In the diagrams of automata over $X = \{0, 1\}$ we then mark active states with σ, leave the label of passive states blank. Figure 1a shows how to draw the Lamplighter automaton of Figure 1b in this way. A few more examples of such diagrams are in Figure 2, and further throughout this paper.



(a) Adding machine, also featured in Figure 4



(b) automaton F



(c) automaton Z

Figure 2: Diagrams of invertible automata

Unless otherwise specified, we assume $X = Y$ everywhere in this text, and write an automaton $\mathcal{A}_q = (S, q, X, \pi, \lambda)$.

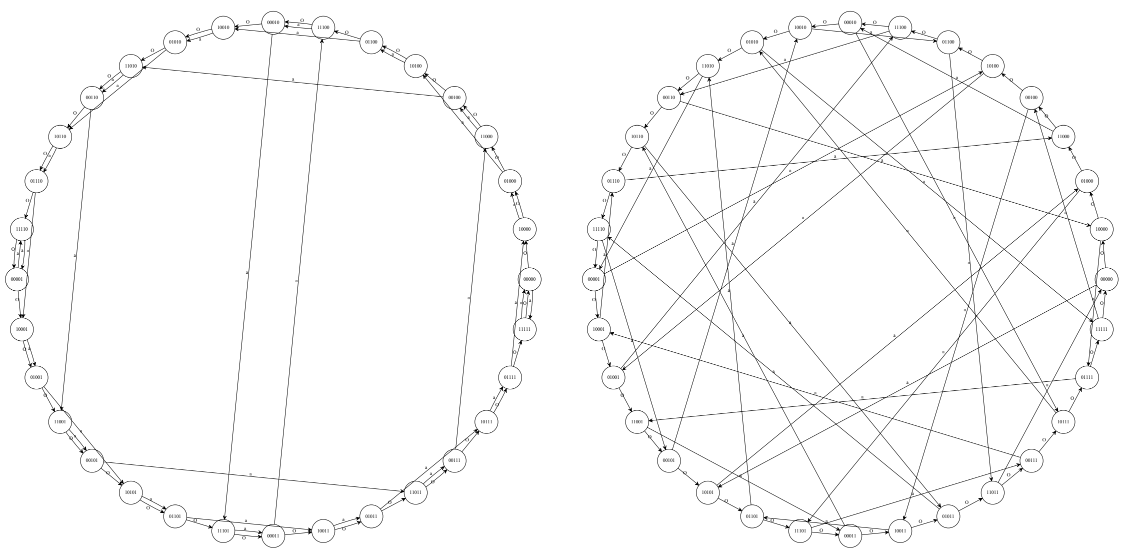
An automaton state q acts on X^*, the d-ary tree $\mathcal{T}$, and its boundary ∂T by the action of $\mathcal{A}_q$. We shall use $\mathcal{A}$ and q interchangeably for this action when the context is clear.

Definition 2.2 *The **graph of the action** of an initial automaton $\mathcal{A}_q = (S, q, X, \pi, \lambda)$ on an invariant subset $S \subset V(\mathcal{T})$ is the directed graph with vertex set S and edges $w \to \lambda_q(w)$ for $w \in S$. The graph of action of automata $\mathcal{A}_{1q_1}, \mathcal{A}_{2q_2}, \ldots, \mathcal{A}_{kq_k}$ on S is similarly defined as a directed graph with vertex set S and edges $w \to \lambda_{iq_i}(w)$, $1 \le i \le k$ and $w \in S$.*

In this paper, we consider **graphs of action on level** n of two automata, $\mathcal{O}$ and $\mathcal{A}$, with $\mathcal{O}$ being the adding machine (Figure 2a). Figure 3 shows examples of such graphs for $\mathcal{A}$ being automaton Z (Figure 2c) and $\mathcal{A}$ being automaton F (Figure 2b).

Definition 2.3 *A Mealy automaton is said to be a **Moore machine** when the output does not depend on the last character of the input. That is, for all $s \in S$, λ_s is constant: for all $x, y \in X$, $\lambda(s, x) = \lambda(s, y)$. In this case, we simply write $\lambda(s)$ for the value λ_s takes.*

Remark 2.4 *In this definition, the output only depends on the current state s. Some authors use the definition of a Moore machine with a shift, where the output is determined by the ending state $\pi(s, x)$, and so does depend on the input.*



(a) The graph of action of adding machine and automaton Z

(b) The graph of action of adding machine and automaton F

Figure 3: Examples of Schreier graphs

Mealy automata $\mathcal{A}$ and $\mathcal{B}$ are said to be **equivalent** if $\mathcal{A}(w) = \mathcal{B}(w)$ for all $w \in X^*$.

Definition 2.5 *An initial Mealy automaton $\mathcal{A}$ is said to be* ***minimal*** *if it has the smallest number of states among all the automata in its equivalence class.*

Minimality is a classical notion, as is the algorithm that produces the minimal automaton in a given class; see [**sholomov**] for a discussion of this algorithm (refer to [**GNS**] for a discussion of this equivalence and a minimization algorithm in the more general case of asynchronous Mealy machines).

Given automata $\mathcal{A}$ and $\mathcal{B}$ such that the output alphabet of $\mathcal{A}$ coincides with the input alphabet of $\mathcal{B}$, one can construct the **product automaton**, denoted $\mathcal{A} \cdot \mathcal{B}$, which computes the composition $\mathcal{A} \circ \mathcal{B}$. We again refer to [**GNS**] for the construction of the product automaton.

2.3 Sections of tree endomorphisms

Definition 2.6 *Let g be an endomorphism of a d-regular rooted tree $\mathcal{T}$, and let w be a finite word. A* ***section*** *of g by w, denoted $g|_w$, is an endomorphism h of $\mathcal{T}$ such that for any word or sequence v, $g(wv) = g(w)h(v)$.*

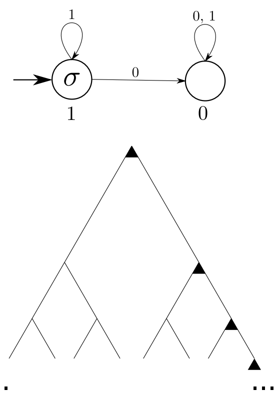


Figure 4: The adding machine and its portrait

Remark 2.7 *a finite automaton $\mathcal{A}$ has only finitely many sections, which correspond to states in the connected component of the starting state in the diagram of the automaton $\mathcal{A}$. More specifically, when g is given by a Mealy machine $\mathcal{A}_q$, $g|_w = \mathcal{A}_{g(w)}$.*

With an invertible tree endomophism g we can associate a **portrait** diagram that uniquely determines g. For a finite word w, $g|_w$ acts on X by a permutation when g is invertible. The portrait consists of the infinite tree $\mathcal{T}$ with markings on the nodes: node corresponding to word w is marked with the permutation of X induced by $g|_w$. When $|X| = 2$, we only mark nodes with the nontrivial permutation and leave others unmarked.

Example 2.8 *The portrait of the adding machine of Figure 2a is shown in Figure 4.* $\triangle$

Every tree automorphism has a portrait, but not all tree automorphisms are given by finite automata. To any tree endomorphism g we can associate a (possibly infinite) automaton $A = (S, g, X, \pi, \lambda)$ with the initial state labeled by g, such that the action of A is identical to the action of g. We take $S = \{g|_w \ : \ w \in X^{\mathbb{N}}\} \cup \{g\}$, and define $\pi(h, x) := h|_x$; $\lambda(h, x) = h(x)$. This automaton of restrictions, in general, need not be finite. When it is finite, the tree automorphism g is said to be **finite-state**.

Remark 2.9 *An automorphism g of the tree $\mathcal{T}$ is finite-state if and only if its portrait contains a finite number of distinct (up to isomorphism of marked trees) subtrees. The subtrees in the portrait diagram define sections of g.*

We now prove several basic propositions related to sections of automorphisms which we use in subsequent chapters. These statements are well-known, but we include them for the reader's convenience.

Proposition 2.10 *If an endomorphism g is invertible, then all of its sections are invertible, and for $w \in X^*$, $(g|_w)^{-1} = g^{-1}|_{g(w)}$.*

Proof. Let $w \in X^*$ and $v \in X^{\mathbb{N}}$. Then by definitions,

$$\begin{aligned} wv &= g^{-1}(g(wv)) \\ &= g^{-1}(g(w)g|_w(v)) \\ &= g^{-1}(g(w))g^{-1}|_{g(w)}(g|_w(v)) \\ &= wg^{-1}|_{g(w)}(g|_w(v)). \end{aligned}$$

Therefore $g^{-1}|_{g(w)}(g|_w(v)) = v$, and the proposition holds. $\square$

Proposition 2.11 *Let g be a tree endomorphism. Then for all w, v finite words w, v over X $g|_{wv} = (g|_w)|_v$.*

Proof. For any word u,

$$g(wvu) = g(w)g|_w(vu) = g(w)g|_w(v)(g|_w)|_v(u) = g(wv)(g|_w)|_v(u).$$

The proposition holds by definition. $\square$

Proposition 2.12 *Let g and h be tree endomorphisms. Then for all finite words w over X, $(gh)|_w = g|_{h(w)}h|_w$.*

Proof. Let v be a finite word. By the definition of section,

$$\begin{aligned} gh(wv) &= g(h(w)h|_w(v)) \\ &= g(h(w))g|_{h(w)}h_w(v), \end{aligned}$$

so $(gh)|_w = g|_{h(w)}h_w$. $\square$

Corollary 2.13 *Let g, h be tree endomorphisms. Then for any finite words w and v, $(gh)|_{wv} = g|_{h(wv)}h|_{wv} = g|_{h(w)h|_w(v)}(h|_w)|_v$.*

Proposition 2.14 $g^n|_w = g|_{g^{n-1}(w)}g|_{g^{n-2}(w)} \dots g|_{g(w)}g|_w$.

Proof. The result holds trivially when $n = 1$. By Proposition 2.12,

$$\begin{aligned} g^n|_w &= (g \circ g^{n-1})|_w \\ &= g|_{g^{n-1}(w)}(g^{n-1}|_w). \end{aligned}$$

The result follows by induction. $\square$

Proposition 2.15 *Assume g acts transitively on levels, $|w| = n$, and $a \in X$. Then $g^{2^n}|_w(a) \neq a$.*

Proof. If $g^{2^n}|_w(a) = a$, then wa, a word of length $n+1$, is a fixed point of g^{2^n}, contrary to the assumption that the length of the orbit of g on words of length $n+1$ is 2^{n+1}.

2.4 Automata with bounded activity

Definition 2.16 *An automaton A is said to have* ***bounded activity*** *if the number of nontrivial sections on every level is bounded by a global constant c:*

$$\exists c : \forall n \in \mathbb{N} : \ |\{A|_w \ : \ A|_w \neq \mathbf{1}, w \in X^n\}| < c.$$

Example 2.17 *The adding machine in Figure 2a has bounded activity.*

This automaton can also be defined by the portrait in Figure 4, in which case it is clear that there is only one nontrivial section on every level. $\triangle$

The following proposition shows that the set of sections of powers of a bounded-activity automaton is finite if the powers are bounded by the number of words on the corresponding level. This fact is used to show Theorem 6.3.

Proposition 2.18 *If A is a tree endomorphism given by a finite Mealy automaton $\mathcal{A}$ which is of bounded activity and acts transitively on levels, then the set*

$$T_A := \{A^n|_w \ : \ w \in X^*, n \leq 2^{|w|}\}$$

is finite.

Proof. by Proposition 2.14,

$$T_A := \{A|_{A^{n-1}(w)}A|_{A^{n-2}(w)} \dots A|_{A(w)}A|_w \ : \ w \in X^*, n \leq 2^{|w|}\}.$$

For a given w, consider a sequence of words $w, A(w), A^2(w), \dots, A^{n-1}(w)$ with $n \leq 2^{|w|}$. By level transitivity of the action of A, all elements in it are distinct words of length $|w|$, and thus this sequence is a subset of vertices on level $|w|$.

Since A is of bounded activity, there is a constant c such that at most c sections on every level are nontrivial. Hence the product

$$A|_{A^{n-1}(w)}A|_{A^{n-2}(w)} \dots A|_{A(w)}A|_w$$

contains at most c nontrivial factor. Since A is finite-state by assumption, its nontrivial sections are enumerated by the finite set of states S_A of $\mathcal{A}$. Therefore, $|T_A| \leq |S_A|^c$. $\square$

2.5 Measure-theoretic definitions

We now give a few definitions relevant to probability theory and ergodic theory.

A **cylinder set** $wX^{\mathbb{N}}$ is a clopen subset of $X^{\mathbb{N}}$ given by

$$wX^{\mathbb{N}} := \{wv \ : \ w \in X^*, v \in X^{\mathbb{N}}\}.$$

A **probability vector** p is a a vector $p : X \to [0,1]$ with $\Sigma_{i \in X} p(i) = 1$. A **stochastic matrix** on X is a matrix $M : X \times X \to [0,1]$ whose rows are probability vectors.

Definition 2.19 *The* ***Bernoulli measure*** *on $X^{\mathbb{N}}$ defined by a probability vector p is given on the cylinders $wX^{\mathbb{N}}$ by*

$$\mu(wX^{\mathbb{N}}) := \prod_{i=0}^{|w|-1} p(w_i),$$

and extended by the additivity properties on all Borel sets. The ***uniform Bernoulli measure*** *is given by $p = \left(\frac{1}{|X|}, \ldots, \frac{1}{|X|}\right)$.*

Informally, this measures probability of a sequence of independent events (e.g. coin flips).

Definition 2.20 *The* ***Markov measure*** *defined by a probability vector $l = (l_x)$ of length $|X|$ and a stochastic matrix $L = (L_{x,y})$ of size $|X| \times |X|$ is given on the cylinder sets $wX^{\mathbb{N}}$ by*

$$\mu(wX^{\mathbb{N}}) := l(w_0) \prod_{i=1}^{|w|-1} L_{w_{i-1}, w_i}.$$

Informally, this measures the probability of events where the probability of an outcome may depend on what the preceding outcome was.

2.6 Sections of a measure

Definition 2.21 *The* ***null measure*** *ν_0 (or the* ***trivial measure****) ν_0 is the measure given by $\nu_0(E) = 0$ for all measurable sets E.*

Definition 2.22 *Suppose μ is a probability measure on $X^{\mathbb{N}}$. If $\mu(wX^{\mathbb{N}}) \neq 0$, then the* ***section*** *of μ by the word $w \in X^*$, denoted $\mu|_w$, is the probability measure on $X^{\mathbb{N}}$ uniquely defined by*

$$\mu|_w(vX^{\mathbb{N}}) := \frac{\mu(wvX^{\mathbb{N}})}{\mu(wX^{\mathbb{N}})}$$

for all $v \in X^*$. In the case $\mu(wX^{\mathbb{N}}) = 0$, we let $\mu|_w$ be the null measure.

The section μ_w can be seen as the conditional probability given w.

For convenience, we also define sections for null measures: if μ is null, $\mu|_w = 0$ for all words w.

We say a word w is **admissible** (with respect to μ) if the section of μ by w is nontrivial (i.e. $\mu(wX^{\mathbb{N}}) \neq 0$). We say a word w is **forbidden** if it is not contained in any admissible word.

Now we describe how to compute sections of measures.

Proposition 2.23 $\mu|_{wv} = (\mu|_w)|_v$ *for all words* $v, w \in X^*$.

Proof.

First, suppose wv is not admissible, i.e. $\mu(wvX^{\mathbb{N}}) = 0$. Then either w is also not admissible, or w is admissible relative to μ, but v is not admissible relative to $\mu|_w$. Either way, $(\mu|_w)|_v$ is the null measure, and the proposition holds.

Now assume $\mu(wvX^{\mathbb{N}}) \neq 0$; then $\mu(wvX^{\mathbb{N}}) \neq 0$. For any word $u \in X^*$ we obtain

$$\begin{aligned}(\mu|_w)|_v(uX^{\mathbb{N}}) &= \frac{\mu|_w(vuX^{\mathbb{N}})}{\mu|_w(vX^{\mathbb{N}})} \\ &= \frac{\mu(wvuX^{\mathbb{N}})}{\mu(wX^{\mathbb{N}})\mu|_w(vX^{\mathbb{N}})} \\ &= \frac{\mu(wvuX^{\mathbb{N}})}{\mu(wvX^{\mathbb{N}})} \\ &= \mu|_{wv}(uX^{\mathbb{N}}). \ \square\end{aligned}$$

Corollary 2.24 *Let* $\mu = \sum_{i=1}^{k} a_i \mu_i$, *where* $a_i \geq 0$ *and* μ_i *are probability measures. The for any admissible word* w,

$$\mu|_w = \frac{1}{\mu\left(wX^{\mathbb{N}}\right)} \sum_{i=1}^{k} a_i \mu_i \left(wX^{\mathbb{N}}\right) \mu_i|_w.$$

Proof. For any word $v \in X^*$,

$$\mu|_w\left(vX^{\mathbb{N}}\right) = \frac{1}{\mu\left(wX^{\mathbb{N}}\right)} \sum_{i=1}^{k} a_i \mu_i \left(wvX^{\mathbb{N}}\right) = \frac{1}{\mu\left(wX^{\mathbb{N}}\right)} \sum_{i=1}^{k} a_i \mu_i \left(wX^{\mathbb{N}}\right) \mu_i|_w \left(vX^{\mathbb{N}}\right). \quad \square$$

3 Finite-state measures

Definition 3.1 *A measure μ is **finite-state** if admits only finitely many distinct sections.*

Example 3.2 *Bernoulli and Markov measures (definitions 2.19 and 2.20, respectively) are finite-state:*

- *any Bernoulli measure μ has only one (nontrivial) section: $\mu|_w = \mu$ whenever w is admissible. Indeed, let p be the defining probability vector,*

$$\begin{aligned}
\mu|_w(vX^{\mathbb{N}}) &= \frac{\mu(wvX^{\mathbb{N}})}{\mu(wX^{\mathbb{N}})} \\
&= \frac{\prod_{i=0}^{|w|-1} p(w_i) \prod_{j=0}^{|v|-1} p(v_j)}{\prod_{i=0}^{|w|-1} p(w_i)} \\
&= \prod_{j=0}^{|v|-1} p(v_j) = \mu(vX^{\mathbb{N}}).
\end{aligned}$$

  Note that if $p(x) = 0$ for some $x \in X$, then words w containing x are not admissible. Conversely, if p is positive, then all words are admissible.

- *a Markov measure μ has at most $|X| + 1$ nontrivial sections: μ (section by the empty word) and $\mu|_x$ for $x \in X$. This is because for all admissible words $w \in X^*$ and all $x \in X$, $\mu|_{wx} = \mu|_x$. Indeed, assuming w is not the empty word, we obtain*

$$\begin{aligned}
\mu|_{wa}(vX^{\mathbb{N}}) &= \frac{\mu(wavX^{\mathbb{N}})}{\mu(waX^{\mathbb{N}})} \\
&= \frac{\Big(l(w_0) \prod_{i=1}^{|w|-1} L(w_{i-1}, w_i)\Big) L(w_{|w|-1}, a) \Big(L(a, v_0) \prod_{j=1}^{|v|-1} L(v_{j-1}, v_j)\Big)}{\Big(l(w_0) \prod_{i=1}^{|w|-1} L(w_{i-1}, w_i)\Big) L(w_{|w|-1}, a)} \\
&= L(a, v_0) \prod_{j=1}^{|v|-1} L(v_{j-1}, v_j) \\
&= \frac{l(a) L(a, v_0) \prod_{j=1}^{|v|-1} L(v_{j-1}, v_j)}{l(a)} \\
&= \frac{\mu(avX^{\mathbb{N}})}{\mu(aX^{\mathbb{N}})} \\
&= \mu|_a(vX^{\mathbb{N}}).
\end{aligned}$$

$\triangle$

Definition 3.3 *A k-**step Markov measure** is a measure μ such that for all words $v \in X^*$ of length k and all words $w \in X^*$, $\mu|_{wv} = \mu|_v$ whenever wv is admissible.*

Informally, this measures the probability of events where the probability of an outcome may depend on what the preceding k outcomes was.

Remark 3.4 *A Markov measure is a 1-step Markov measure. A k-step Markov measure on $X^{\mathbb{N}}$ with $|X| = d$ is finite-state with at most $\dfrac{d^{k+1}-1}{d-1}$ sections.*

Indeed, a finite d-tree of depth $k+1$ has $1+d+d^2+\ldots+d^k = \frac{d^{k+1}-1}{d-1}$ nodes, which encode all words of length not exceeding k. By definition, every nontrivial section of a k-step Markov measure is a section by one of these words. $\square$

Definition 3.5 *To any finite-state measure μ we associate an automaton $\mathcal{A}_\mu$ as follows.*

Let $\mu_1, \ldots, \mu_n$ be the distinct sections of μ. Consider an automaton $\mathcal{A}_\mu$ with input alphabet X, output alphabet $Y \subset [0,1]$, state set $S = \{\mu_1, \ldots, \mu_n\}$, initial state $s_0 = \mu \in S$, and transition and output functions defined by

$$\begin{aligned} \pi(\mu_i, a) &:= \mu_i|_a; \\ \lambda(\mu_i, a) &:= \mu_i(aX^{\mathbb{N}}). \end{aligned} \tag{1}$$

*We say that $\mathcal{A}_\mu$ **determines** the measure μ.*

Proposition 3.6 *The automaton $\mathcal{A}_\mu$ uniquely determines μ as follows: for any input word $w \in X^*$, the output word $\mathcal{A}_\mu(w) = p_0p_1\ldots p_{|w|-1}$ is a sequence of real numbers whose product is $\mu(wX^{\mathbb{N}})$:*

$$\mu(wX^{\mathbb{N}}) = \prod_{i=0}^{|w|-1} (\mathcal{A}_\mu(w))_i. \tag{2}$$

Proof. The proposition holds for when $|w| = 1$ by construction; assume it holds for all words of length k. Consider an arbitrary word $w = w_0w_1\ldots w_k$ of length k. Then applying the inductive hypothesis, and then applying Proposition 2.23 k times, we obtain:

$$\begin{aligned} \prod_{i=0}^{k} p_i &= \mu(w_0w_1\ldots w_{k-1}X^{\mathbb{N}}) \cdot ((\ldots(\mu|_{w_0})|_{w_1})|_{w_2})\ldots)|_{w_{k-1}}(w_kX^{\mathbb{N}}) \\ &= \mu(w_0w_1\ldots w_{k-1}X^{\mathbb{N}})\mu|_{w_0w_1\ldots w_{k-1}}(w_kX^{\mathbb{N}}) \\ &= \mu(w_0w_1\ldots w_kX^{\mathbb{N}}) = \mu(wX^{\mathbb{N}}), \end{aligned}$$

which completes the inductive step. $\square$

Note that if μ_i is a section of μ, then the automaton defining μ_i can be obtained from $\mathcal{A}_\mu$ by changing the initial state to μ_i and possibly dropping some states (as some sectinos of μ might not be sections of μ_i).

Definition 3.7 *Suppose μ is a finite-state measure that admits a trivial section. We call the corresponding state of the defining automaton $\mathcal{A}_\mu$ trivial.*

Refer to Example 3.14 for an automaton with a trivial state; for example, the state $\mu|_{11}$ in Figure 8a is trivial.

Remark 3.8 *Given a finite-state measure μ, the automaton defined in 1 is minimal and contains at most one trivial state.*

Example 3.9 *The automaton computing a Bernoulli measure on $\{0,1\}^{\mathbb{N}}$ defined by a positive probability vector $p = (p(0), p(1))$ is depicted in Figure 5a.* $\triangle$

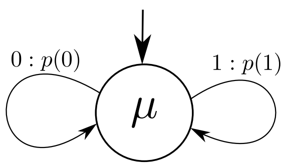
Example 3.10 *The automaton computing a Markov measure on $\{0,1\}^{\mathbb{N}}$ defined by a positive probability vector $l = (l(0), l(1))$ and a positive probability matrix $L = (L_{ij})$ is depicted in Figure 5b.* $\triangle$

Example 3.11 *Figure 6 shows a general 2-step Markov measure on $\{0,1\}^{\mathbb{N}}$. Such a measure is determined by a probability vector p, a stochastic matrix q and a probability tensor M (M_{ijk} gives the probability of k given ij).* $\triangle$

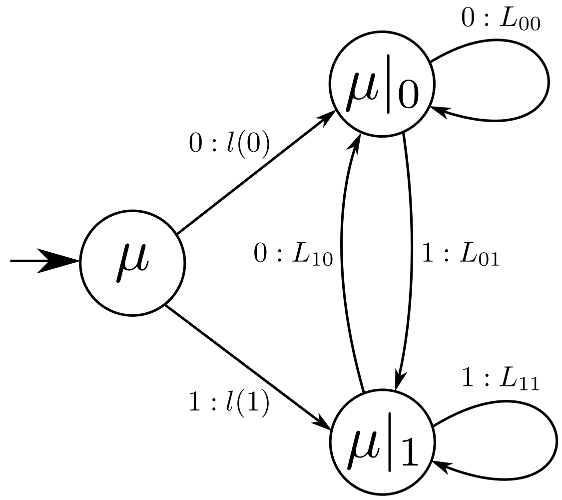
Similarly to tree automorphisms, we define the **portrait of the measure** μ to be the diagram consisting of the marked tree $\mathcal{T}$, where the node corresponding to a word w is marked with the values $\mu|_w$ takes on cylinders $xX^{\mathbb{N}}$, $x \in X$. A portrait defines a measure uniquely.

When dealing with probability measures, it is often convenient to consider the vector $p_w := \left(\mu|_w(x_0X^{\mathbb{N}}), \ldots, \mu|_w(x_{d-1}X^{\mathbb{N}}\right)$ up to scaling. Since $\sum_{i=0}^{d-1} \mu(x_iX^{\mathbb{N}}) = 1$, the **proportion** $p_{w_0} : p_{w_1} : \ldots : p_{w_{n-1}} \in \mathbb{R}P^{d-1}$ determines the values of μ on X unambiguously. We then use the proportion as the corresponding label in the portrait.

Example 3.12 *The uniform Bernoulli measure on a binary alphabet has one section whose proportion is $1:1$. Its portrait is shown in Figure 7.* $\triangle$



(a) Diagram of the automaton computing a Bernoulli measure



(b) Diagram of the automaton computing a Markov measure

Figure 5: Automata determining a Bernoulli and a Markov measure on $\{0,1\}^{\mathbb{N}}$

Remark 3.13 *As with automorphisms, one can draw the portrait of any probability measure on the space $X^{\mathbb{N}}$, but not all probability measures are finite-state.*

Again, as with automorphisms, even small automata define interesting finite-state measures.

It should be noted that even small automata define interesting finite-state measures.

Example 3.14 *The measure μ defined by the automaton in Figure 8a is a 2-step Markov measure on $\Omega = \{0,1\}^{\mathbb{N}}$ that is not a 1-step Markov measure on Ω. It is supported on the* ***Fibonacci subshift****, which is the (shift-invariant) subset of Ω consisting of all sequences that do not contain consecutive 1's. The number of nontrivial sections of μ by words of length $n-1$ is the n'th Fibonacci number as can be seen in the portrait of μ shown in Figure 8b (to simplify the figure, we omitted the subtrees corresponding to the null measure).* $\triangle$

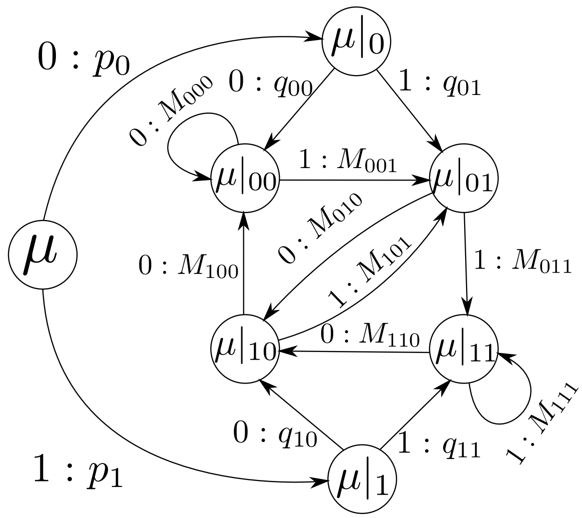


Figure 6: Automaton defining a general 2-step Markov measure on $\{0,1\}^{\mathbb{N}}$

4 Images of finite-state measures under tree automorphisms

Given a finite-state measure μ and a tree automorphism g, we consider the pushforward measure $g_*\mu$ defined by $g_*\mu(E) = \mu(g^{-1}(E))$ for all measurable sets E. We say that $g_*\mu$ is the image of μ under (the map) g.

The following proposition is useful for constructing the automata of finite-state measures which are images under automaton automotphisms.

Proposition 4.1 *Let $A = (X, S, s_0, \pi, \lambda)$ be a Mealy automaton with initial state $s_0 = g$ acting on $\mathcal{T}$, and let ν be a probability measure on $\partial\mathcal{T}$. Then for $x \in X$,*

$$(g_*\nu)(xX^{\mathbb{N}}) = \sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}});$$

$$(g_*\nu)|_x = \frac{\sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}})\pi_g(y)_*(\nu|_y)}{\sum_{y \in \lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}})}.$$

where $\pi_g(x) := \pi(g, x)$ and $\lambda_g(x) := \lambda(g, x)$.

Proof. Note that for a word $w \in X^*$,

$$g^{-1}(xwX^{\mathbb{N}}) = \bigsqcup_{y \in \lambda_g^{-1}(x)} y\pi_g(y)^{-1}(wX^{\mathbb{N}}).$$

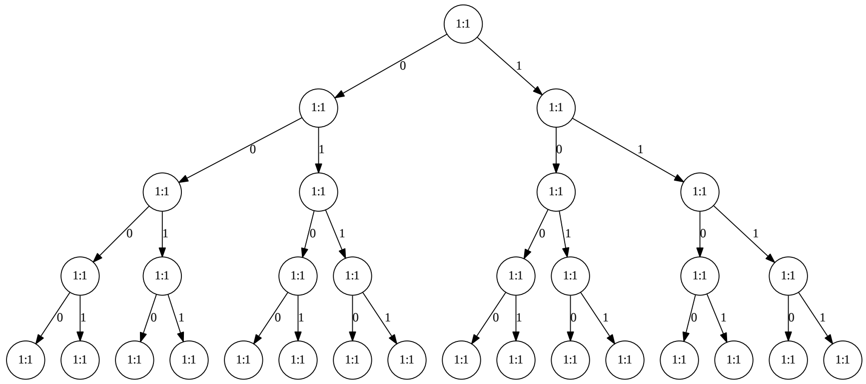


Figure 7: Portrait of the uniform Bernoulli measure up to level 5

By definition,

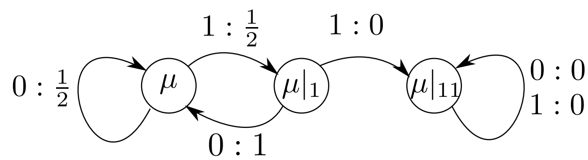
$$\begin{aligned}(g_*\nu)|_x(wX^{\mathbb{N}}) &= \frac{(g_*\nu)(xwX^{\mathbb{N}}}{(g_*\nu)(xX^{\mathbb{N}})}\\ &= \frac{\nu(g^{-1}(xwX^{\mathbb{N}})}{\nu(g^{-1}(xX^{\mathbb{N}}))}\\ &= \frac{\sum\limits_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}})\nu|_y\left(\pi_g(y)^{-1}(wX^{\mathbb{N}})\right)}{\sum\limits_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}})\nu|_y\left(\pi_g(y)^{-1}(X^{\mathbb{N}})\right)}\\ &= \frac{\sum\limits_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}})\pi_g(y)_*(\nu|_y)(wX^{\mathbb{N}})}{\sum\limits_{y\in\lambda_g^{-1}(x)} \nu(yX^{\mathbb{N}})}. \square\end{aligned}$$

Corollary 4.2 *When g is as in Prop. 4.1, and ν is a Bernoulli measure given by probability vector p, then its image under g satisfies*

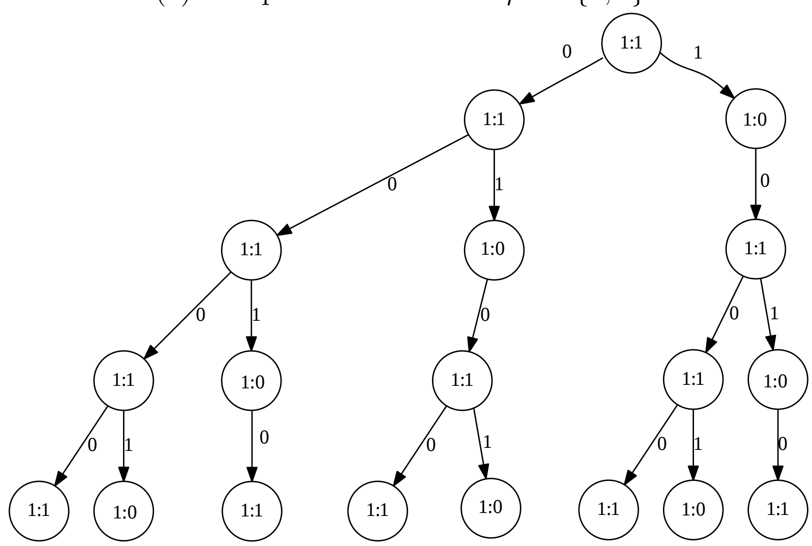
$$(g_*\nu)|_x = \frac{\sum\limits_{y\in\lambda_g^{-1}(x)} p(y)\pi_g(y)_*(\nu)}{\sum\limits_{y\in\lambda_g^{-1}(x)} p(y)}.$$

In particular, when ν is uniform Bernoulli, $(g_*\nu)(xX^{\mathbb{N}}) = |\lambda_g^{-1}(x)|/|X|$, and

$$(g_*\nu)|_x = \frac{1}{|\lambda_g^{-1}(x)|}\sum_{y\in\lambda_g^{-1}(x)} \pi_g(y)_*(\nu).$$



(a) 2-step Markov measure μ on $\{0, 1\}^{\mathbb{N}}$



(b) Portrait of μ up to level 5

Figure 8: A finite-state measure supported on the Fibonacci subshift

When ν is uniform Bernoulli, its pushforwards by invertible endomorphisms are also uniform Bernoulli:

Proposition 4.3 *When ν is uniform Bernoulli and g is invertible, $g_*\nu = \nu$.*

Proof. For $w \in X^*$,

$$g_*\nu(wX^{\mathbb{N}}) = \nu(g^{-1}(wX^{\mathbb{N}}) = \nu(g^{-1}(w)X^{\mathbb{N}}) = |X|^{-|w|} = \nu(wX^{\mathbb{N}}). \square$$

5 Log map

Let A be an automorphism of the d-regular rooted tree $\mathcal{T}$ that acts transitively on each level. Recall that level n of the tree consists of d^n words of length n. Hence for any pair of words w_1, w_2 of length n, there is a unique integer k, $0 \leq k \leq d^n - 1$ such that $A^k(w_1) = w_2$. Furthermore, if $A^k(w_1) = A^{k'}(w_1)$ for some integers k and k', then $k \equiv k' \mod d^n$.

Definition 5.1 *For any $n \geq 1$, the* ***displacement*** *function $\boldsymbol{d}_{A,n} : X^n \times X^n \to \mathbb{Z}/d^n\mathbb{Z}$ is defined on pairs of words w_1, w_2 of length n by*

$$\boldsymbol{d}_{A,n}(w_1, w_2) := [k]_{d^n},$$

where $A^k(w_1) = w_2$ and $[k]_{d^n} \in \mathbb{Z}/d^n\mathbb{Z}$ is the equivalence class mod d^n. We write $[k]$ when n is understood from the context.

Definition 5.2 *For any integers m and n, $1 \leq m \leq n$, the* ***natural projection*** *$\phi_{m,n} : \mathbb{Z}/d^n\mathbb{Z} \to \mathbb{Z}/d^m\mathbb{Z}$ is defined by $\phi_{m,n}([k]_{d^n}) := [k]_{d^m}$. These functions are homomorphisms of rings.*

The functions $\mathbf{d}_{A,n}$ for different values of n are compatible with each other with respect to the natural projections.

Proposition 5.3 *Suppose $|w_1| = |w_2| = n$ and $a, b \in X$. Then*

$$\phi_{n,n+1}(\boldsymbol{d}_{A,n+1}(w_1 a, w_2 b)) = \boldsymbol{d}_{A,n}(w_1, w_2).$$

Proof: Let $\mathbf{d}_{A,n}(w_1, w_2) = [k]$ so that $A^k(w_1) = w_2$, with $0 \leq k \leq d^n - 1$. Let $a' = A^k|_{w_1}(a)$. Then $A^k(w_1 a) = w_2 a'$. Note that

$$\begin{aligned} A^{d^{n+k}}(w_1 a) &= A^{d^n}(w_2 a') \\ &= A^{d^n}(w_2) A^{d^n}|_w(a'). \end{aligned}$$

By Proposition 2.15,

$$A^{d^n}|_w(a'), A^{2d^n}|_w(a'), \ldots, A^{(k-1)d^n}|_w(a'),$$

are all distinct. Since $|X| = d$, this implies $A^{td^n}|_w(a') = b$ for some t, $0 \leq t \leq d-1$. Thus $A^{td^n+k}(w_1 a) = w_2 b$, whence $d_{A,n+1}(w_1 a, w_2 b) = [k + td^n]$.

Since $\phi_{n,n+1}([k + td^n]) = [k]$, the proposition holds. $\square$.

In addition to the tree endomorphism A, let us consider a tree endomorphism B.

Definition 5.4 *For any $n \geq 1$, $\operatorname{Log}_{A,n}(B) : X^n \to \mathbb{Z}/d^n\mathbb{Z}$ is a function which calculates the displacement of a word w of length n along the orbit of A under the action of B:*

$$\operatorname{Log}_{A,n}(B)(w) := \boldsymbol{d}_{A,n}(w, B(w)).$$

Note that for any word w of length n, $A^{\operatorname{Log}_{A,n}(B)}(w) = B(w)$, which motivates the name "logarithm" for this function.

Corollary 5.5 *For any word w of length n and character $x \in X$,*

$$\phi_{n,n+1}(\mathrm{Log}_{A,n+1}(B)(wa)) = \mathrm{Log}_{A,n}(B)(w).$$

In other words, the displacement of wa by B along the orbit of A is either the same as displacement of w or differs from it by a multiple of d^n.

Corollary 5.6 *For any integers m and n, $1 \leq m < n$, the following diagram commutes:*

$$\begin{tikzcd} X^n \arrow[r, "\sigma_r^{n-m}"] \arrow[d, "\mathrm{Log}_{A,n}(B)"'] & X^m \arrow[d, "\mathrm{Log}_{A,m}(B)"'] \\ \mathbb{Z}/d^n\mathbb{Z} \arrow[r, "\phi_{m,n}"] & \mathbb{Z}/d^m\mathbb{Z} \end{tikzcd}$$

Here, σ_r is the operator that trims the word, deleting the least letter: $\sigma(wa) = w$ for any word w and a character $a \in X$.

Proof. This follows from Corollary 5.5 by induction on $n - m$. $\square$

Let $\mathbb{Z}_d$ be the inverse limit of the directed system

$$\mathbb{Z}/d^n\mathbb{Z} \xrightarrow{\phi_{m,n}} \mathbb{Z}/d^m\mathbb{Z}$$

(for $m, n \in \mathbb{N}$). $\mathbb{Z}_d$ comes with a natural structure of a ring, and is known as the ring of the d-adic integers (note that d need not be prime).

Since the boundary of the tree $\partial\mathcal{T}$ can also be seen as the inverse limit of the directed system

$$X^n \xrightarrow{\sigma_r^{n-m}} X^m,$$

Corollary 5.6 implies that there exists a unique function $\mathrm{Log}_A(B) : \partial\mathcal{T} \to \mathbb{Z}_d$, which restricts to $\mathrm{Log}_{A,n}(B)$ on level n for all n.

Definition 5.7 *The logarithm $\mathrm{Log}_A(B)$ is the inverse limit*

$$\mathrm{Log}_A(B) = \varprojlim_n \mathrm{Log}_{A,n}(B).$$

That is, it is the unique function $\mathrm{Log}_A(B) : \partial\mathcal{T} \to \mathbb{Z}_d$ that makes the following diagram commute:

$$\begin{tikzcd} \partial\mathcal{T} \arrow[r, "\pi_n"] \arrow[d, "\mathrm{Log}_A(B)"'] & X^n \arrow[d, "\mathrm{Log}_{A,n}(B)"'] \\ \mathbb{Z}_d \arrow[r, "\pi_n"] & \mathbb{Z}/d^n\mathbb{Z} \end{tikzcd}$$

(π_n are the natural projections of the corresponding inverse limits).

Any positive integer N admits a unique d-ary expansion

$$N = \sum_{i=0}^{k} a_i d^i,$$

where each $0 \le a_i \le d-1$. This way, the set $\mathbb{Z}/d^n\mathbb{Z}$ can be identified with the set of words of length n over the alphabet X. Consequently, the set $\mathbb{Z}_d$ can be identified with infinite words in alphabet X, which, in term, are identified with the boundary of the tree $\partial\mathcal{T}$. Therefore, $\mathrm{Log}_{A,n}(B)$ can be seen as a transformation of the n'th level, and $\mathrm{Log}_A(B)$ can be regarded as a transformation of $\partial\mathcal{T}$.

Proposition 5.8 *There exists an endomorphism of the tree $\mathcal{T}$ such that $\mathrm{Log}_{A,n}(B)$ is the restriction of the endomorphism to level n, and $\mathrm{Log}_A(B)$ is the action of the endomorphism on the boundary $\partial\mathcal{T}$.*

Proof. Let $L : X^* \to X^*$ be the transformation that coincides with $\mathrm{Log}_{A,n}(B)$ on level n for all n. By construction, L preserves the levels. Corollary 5.5 implies that L maps adjacent vertices of $\mathcal{T}$ to adjacent vertices. Therefore, L is an endomorphism. By Definition 5.7, the action of L on the boundary $\partial\mathcal{T}$ is exactly $\mathrm{Log}_A(B)$. $\square$

In the rest of the paper, we deal with $d = 2$, and so identify (and use interchangeably) the dyadic numbers and infinite binary sequences (elements of $\partial\mathcal{T}$).

Remark 5.9 *The construction of the logarithm map $\mathrm{Log}_A(B)$ (including Proposition 5.8) can be extended from d-regular trees to spherically homogeneous trees (defined in, e.g., [**BORT**]).*

6 The automaton computing the Log map

Here and onwards we assume that $X = \{0, 1\}$, that is, $\mathcal{T}$ is the binary rooted tree.

Let A be an automorphism of the tree that acts transitively on each level, and let B be an endomorphism. In light of Proposition 5.8, the Log map $\mathrm{Log}_A(B)$ can be regarded as an endomorphism of $\mathcal{T}$.

To simplify notation, we will denote by Log both $\mathrm{Log}_A(B)$ and $\mathrm{Log}_{A,n}(B)$.

In this section we construct an automaton which computes this endomorphism.

We further assume that the automorphism A is of bounded activity (in the sense of Definition 2.16). An example of such endomorphism is the adding machine, whose automaton is shown in Figure 2a.

Remark 6.1 *Any tree automorphism that acts transitively on levels is conjugate to the adding machine.*

The assumption that A is of bounded activity allows us to prove the following lemma, which will be useful in the construction of the automaton for Log.

Lemma 6.2 *If A, B are tree endomorphisms given by finite automata, A is of bounded activity and acts transitively on all levels, then the set $S_{A,B}$ consisting of triples of sections:*

$$S_{A,B} := \{(B|_w, A^{d(w)}|_w, A^{2^{|w|}}|_w) \ : \ w \in X^*\}$$

is finite.

Proof. By Proposition 2.18, the set

$$T_A := \{A^n|_w \ : \ w \in X^*, n \leq 2^{|w|}\}$$

is finite. Note that $A^{d(w)}|_w, A^{2^{|w|}}|_w \in T_A$ for all $w \in X^*$. Let S_B be the set of states of the automaton of B. Then

$$|S_{A,B}| \leq |S_B| \cdot |T_A|^2. \ \square$$

The set $S_{A,B}$ is going to be the set of states of our automaton. See Example 6.5 for an explicit computation of $S_{A,B}$.

Theorem 6.3 *Let A, B be as above. Consider the automaton $\mathcal{L} = \mathcal{L}_{A,B}$ with set of states $S_{A,B}$, initial state $(B, \mathbf{1}, A)$ (where $\mathbf{1}$ is the identity automorphism), transition function π defined by*

$$\begin{aligned}\pi((\beta,\gamma,\delta),a) &:= (\beta',\gamma',\delta'), \textit{ where}\\ \beta' &= \beta|_a\\ \gamma' &= \begin{cases}\gamma|_a & \textit{if } \beta(a)=\gamma(a);\\ (\gamma\delta)|_a, & \textit{otherwise};\end{cases}\\ \delta' &= \delta^2|_a\end{aligned}$$

and the output function λ given as follows:

$$\lambda((\beta,\gamma,\delta),a) := \begin{cases}0, & \textit{if } \beta(a)=\gamma(a);\\ 1, & \textit{otherwise.}\end{cases}$$

Then the transition function is well-defined, and the automaton $\mathcal{L}$ outputs $\mathrm{Log}_{A,n}(B)$ *as a dyadic integer:*

$$\mathrm{Log}_{A,n}(B)(w) = \sum_{i=0}^{|w|-1} \mathcal{L}(w)_i 2^i, \tag{3}$$

where $n = |w|$, $\mathrm{Log}_{A,n}(B)$ *is the displacement function in Definition 5.4, and* $\mathcal{L}(w)_i$ *is the* i*'th character of the word* $\mathcal{L}(w)$*.*

Proof. We first show that upon reading a word w, the automaton L ends up in the state

$$\left(B|_w, A^{d(w)}|_w, A^{2^{|w|}}|_w\right) \in S_{A,B}.$$

This hypothesis holds for the empty word. We proceed by induction on $|w|$.

Assume the hypothesis holds for all $|w| \leq n$.

To prove the inductive hypothesis for words of length $n+1$, let $|w| = n$ and $a \in X$, and assume that $\mathcal{L}$ is in the state (β, γ, δ) after reading w. We show that

$$(\beta', \gamma', \delta') := \pi((\beta, \gamma, \delta), a) = \left(B|_{wa}, A^{d(wa)}|_{wa}, A^{2^{|wa|}}|_{wa}\right).$$

Indeed:

1. $\beta' = \beta_a$ by definition, and
$$\begin{aligned} B|_{wa} &= (B|_w)|_a \quad \text{(by Proposition 2.11)} \\ &= \beta_a \\ &= \beta'. \end{aligned}$$

2. Note that $A^{2^{|w|}}(w) = w$ by transitivity of A. By definition, $\delta' = \delta^2|_a = \delta|_{\delta(a)}\delta|_a$. Now
$$\begin{aligned} A^{2^{|wa|}}|_{wa} &= A^{2^{|w|+1}}|_{wa} \\ &= (A^{2^{|w|}})^2|_{wa} \\ &= A^{2^{|w|}}|_{A^{2^{|w|}}(wa)} A^{2^{|w|}}|_{wa} \quad \text{(by Proposition 2.12)} \\ &= A^{2^{|w|}}|_{A^{2^{|w|}}(w)A^{2^{|w|}}|_w(a)} (A^{2^{|w|}}|_w)|_a \\ &= A^{2^{|w|}}|_{w\delta(a)}\delta|_a \quad \text{(by Proposition 2.11, inductive assumption, and } A^{2^{|w|}} = w) \\ &= (A^{2^{|w|}}|_w)|_{\delta(a)}\delta|_a \\ &= \delta|_{\delta(a)}\delta|_a \\ &= \delta'. \end{aligned}$$

3. Let $d(w) := \mathrm{Log}_{A,n}(B)(w)$. By definition of $\mathrm{Log}_{A,n}(B)$, $B(w) = A^{d(w)}(w)$. Note that

$$B(wa) = B(w)B|_w(a) = A^{d(w)}(w)\beta(a);$$
$$A^{d(w)}(wa) = A^{d(w)}(w)A^{d(w)}|_w(a) = A^{d(w)}(w)\gamma(a).$$

If $\beta(a) = \gamma(a)$, then $B(wa) = A^{d(w)}(wa)$, and thus $d(wa) = d(wa) = d(w)$ by definition of $d = \mathrm{Log}_{A,n}(B)$. Otherwise, $d(wa) = d(w) + 2^{|w|}$ since this is the only other possibility. Therefore,

$$A^{d(wa)} = \begin{cases} A^{d(w)}, \text{ if } \beta(a) = \gamma(a); \\ A^{d(w)}A^{2^{|w|}}, \text{ otherwise.} \end{cases}$$

Now we compute:

$$\begin{aligned} A^{d(w)}|_{wa} &= (A^{d(w)}|_w)|_a \\ &= \delta|_a; \\ A^{d(w)}A^{2^{|w|}}|_{wa} &= A^{d(w)}|_{A^{2^{|w|}}(wa)}(A^{2^{|w|}}|_w)|_a \\ &= A^{d(w)}|_{A^{2^{|w|}}(w)A^{2^{|w|}}|_w(a)}\delta|_a \\ &= A^{d(w)}|_{w\delta(a)}\delta|_a \\ &= (A^{d(w)}|_w)|_{\delta(a)}\delta|_a \\ &= \gamma|_{\delta(a)}\delta|_a \\ &= (\gamma\delta)|_a \end{aligned}$$

Therefore

$$A^{d(wa)}|_{wa} = \begin{cases} \gamma|_a \text{ if } \beta(a) = \gamma(a); \\ (\gamma\delta)|_a \text{ otherwise.} \end{cases}$$

This matches the definition of γ', and thus $\gamma' = A^{d(wa)}|_{wa}$.

In particular, we have verified that the transition function π is well-defined, since its values are always in the set $S_{A,B}$.

This completes the proof of the hypothesis that the automaton is in state $\left(B|_w, A^{d(w)}|_w, A^{2^{|w|}}|_w\right)$ after reading w.

Furthermore, we observed that

$$d(wa) = \begin{cases} d(w), \text{ if } \beta(a) = \gamma(a); \\ d(w) + 2^{|w|} \text{ otherwise.} \end{cases}$$

From this observation and the definition of λ, equation 3 follows by induction.

This completes the proof of the theorem. $\square$

Proposition 6.4 *When A and B are as in Theorem 6.3 and, additionally, B is invertible, the automaton $L_{A,B}$ is a Moore machine (as in Definition 2.3). Recall that the value of the output function $\lambda(s, x)$ of a Moore machine only depends on the state s.*

Proof. By assumption, A is invertible, and so is $A^{d(w)}$ for any $w \in X^*$. B is invertible by assumption. By Proposition 2.10, their sections $\beta = B|_w$ and $\gamma = A^{d(w)}|_w$ are invertible, and so is $\beta\gamma^{-1}$.

Now the set of permutations $\mathrm{Perm}(\{0,1\}) = \{\mathbf{1}, \sigma\}$, so either $\beta\gamma^{-1}(x) = (x)$, or $\beta\gamma^{-1}(x) = \sigma(x)$.

In the first case, $\lambda(\beta, \gamma, \delta)(x) = 0$ for $x \in \{0, 1\}$.

Otherwise, since permutation σ has no fixed points, $\beta(x) \neq \gamma(x)$ and $\lambda(\beta, \gamma, \delta)(x) = 1$ for $x \in \{0, 1\}$. $\square$

Example 6.5 *Let A be the **adding machine** (see Figure 2a) with states A and $\mathbf{1}$ (trivial state). Let automaton F be given by Figure 2b and have states $\{a, b, c\}$, with initial state a. We consider $\mathrm{Log}_A F$.*

Note that

$$A^2|_a = A|_A(a)A|_a = A,$$

since $A|_0A|_1 = A|_1A|_0 = A$. Therefore, $A^{2^{|w|}}|_w = A$ for all $w \in X^*$ (intuitively, adding 2^n to a dyadic number is the same as adding 1 to $n+1$'st digit).

We thus have $S_{A,B} \subset \{a, b, c\} \times \{A, 1\} \times \{A\}$. Consequently, $|S_{A,B}| \leq 6$.

*Let us compute the transition and the output function for $L_{A,B}$. By Proposition 6.4, $L_{A,B}$ is a Moore machine, so we let * stand for either 0 or 1 in what follows:*

$$\begin{array}{ccc} \lambda((a,1,A),*) = 1 & \lambda((b,1,A),*) = 1 & \lambda((c,1,A),*) = 0 \\ \lambda((a,A,A),*) = 0 & \lambda((b,A,A),*) = 0 & \lambda((c,A,A),*) = 1 \end{array}$$

We can use this to compute the transition function:

$$\begin{array}{ccc} \pi((a,1,A),0) = (c,1,A) & \pi((b,1,A),0) = (b,1,A) & \pi((c,1,A),0) = (a,1,A) \\ \pi((a,1,A),1) = (b,A,A) & \pi((b,1,A),1) = (c,A,A) & \pi((c,1,A),1) = (a,1,A) \\ \pi((a,A,A),0) = (c,1,A) & \pi((b,A,A),0) = (b,1,A) & \pi((c,A,A),0) = (a,A,A) \\ \pi((a,A,A),1) = (b,A,A) & \pi((b,A,A),1) = (c,A,A) & \pi((c,A,A),1) = (a,A,A) \end{array}$$

Since $\delta = A$ for all $(\beta, \gamma, \delta) \in S_{A,B}$, we omit it and write (β, γ) for (β, γ, A) in $\mathcal{L}_{A,B}$. The automaton $\mathcal{L}_{A,B}$ we have computed here is in in Figure 9. $\triangle$

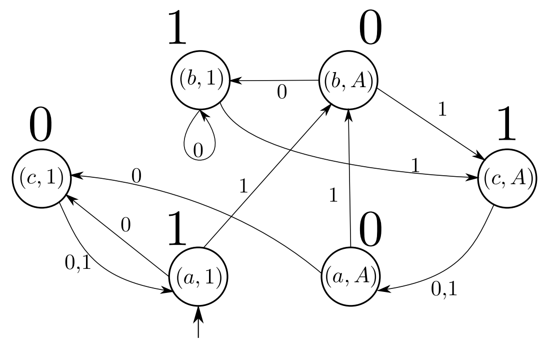


Figure 9: Automaton $L_{A,B}$ when A is the adding machine and B is automaton F. The output from a state is the big number next to it.

Example 6.5 calls for a more efficient notation in the case when A is the adding machine and B is invertible:

Corollary 6.6 *Let A be the* ***adding machine*** *given by automaton of Figure 2a, and assume B is invertible. Then $\delta = A$ for all (β, γ, δ) in the connected component of $(B, \boldsymbol{1}, A)$ in $L_{A,B}$, and so can be omitted. After relabeling $(\beta, \gamma, \delta) \to (\beta, \gamma)$ in $\mathcal{L}_{A,B}$, we obtain the Moore machine $\hat{\mathcal{L}}_{A,B}$ with initial state $(B, \boldsymbol{1})$, and transition and output functions π and λ as specified in Table 1.*

Note: $\hat{\mathcal{L}}$ and $\mathcal{L}$ are equivalent automata: $\mathcal{L}(w) = \hat{\mathcal{L}}(w)$ for all words w.

	β and γ are both active or both passive	Exactly one of β and γ is active
$\pi((\beta, \gamma), a)$	$(\beta\|_a, \gamma\|_a)$	$(\beta\|_a, (\gamma A)\|_a)$
$\lambda((\beta, \gamma))$	0	1

Table 1: Transition and output functions of the automaton computing $\mathrm{Log}_A(B)$ when A is the adding machine and B is invertible

Proof. Observe that

$$A^2|_a = A|_A(a)A|_a = A,$$

since $A|_0A|_1 = A|_1A|_0 = A$. Since the initial state is $(B, \mathbf{1}, A)$, it follows that the rest of the states in the connected component of $L_{A,B}$ containing the initial state are of the form (β, γ, A). Similarly, $\gamma \in \{\mathbf{1}, A\}$.

The rest follows from the construction 6.3 and Prop. 6.4. Note that $\beta(x) = \gamma(x)$ for $x \in X = \{0, 1\}$ if and only if β and γ are both active or both passive. $\square$

When A, B are as in the Corollary above, it is easy to construct $\mathcal{L}_{A,B}$, since once can see β, γ are active or passive by examining the diagram of the automatons B and A.

Remark 6.7 *When B is invertible, and $\beta \in S(B)$ is a state of B, the transition function λ of B at β, λ_β, takes values in* $\mathrm{Perm}(X) = \{\mathbf{1}, \sigma\}$. *The Table 1 of Proposition 6.6 can be rewritten out explicitly as Table 2.*

λ_β	γ	x	$\pi((\beta,\gamma),x)$	$\lambda((\beta,\gamma),x)$
$\mathbf{1}$	$\mathbf{1}$	0	$(\pi(\beta,0),\mathbf{1})$	0
$\mathbf{1}$	$\mathbf{1}$	1	$(\pi(\beta,1),\mathbf{1})$	0
σ	A	0	$(\pi(\beta,0),\mathbf{1})$	0
σ	A	1	$(\pi(\beta,1),A)$	0
$\mathbf{1}$	A	0	$(\pi(\beta,0),A)$	1
$\mathbf{1}$	A	1	$(\pi(\beta,1),A)$	1
σ	$\mathbf{1}$	0	$(\pi(\beta,0),\mathbf{1})$	1
σ	$\mathbf{1}$	1	$(\pi(\beta,1),A)$	1

Table 2: Table 1 with explicit values of π, γ

Example 6.8 *We compute the distance automaton when A is the adding machine, and B is the Bellaterra automaton (Figure 10). This automaton is so called because it was was studied during the summer school in Automata Groups in the Autonomous University of Barcelona in Bellaterra. An interesting property of it is that the group generated by its states is a free product of 3 copies of $\mathbb{Z}/2\mathbb{Z}$ [***SavVor***].*

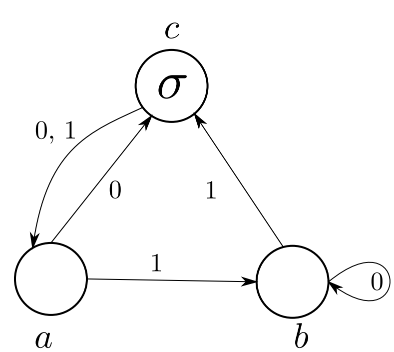


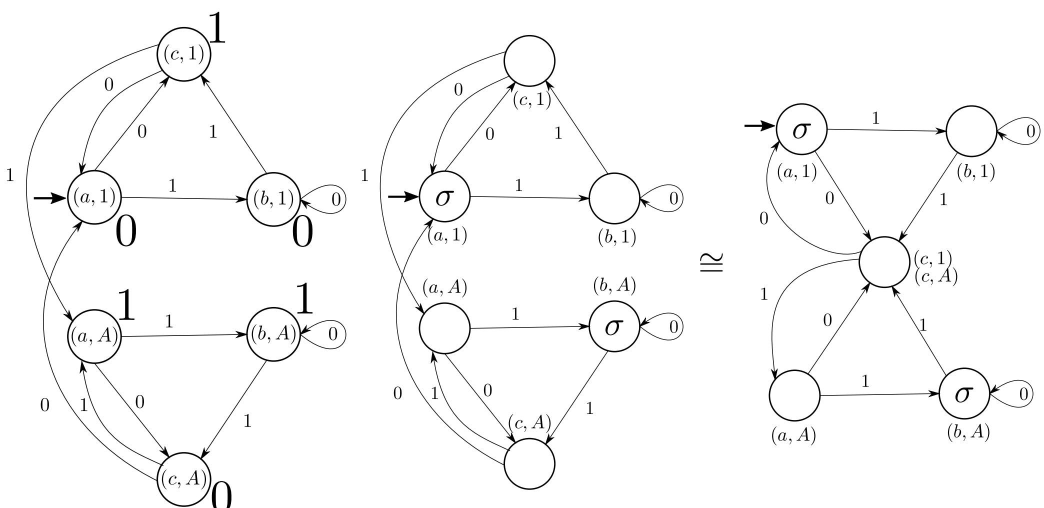
Figure 10: Bellaterra automaton

Using the new notation:

$$
\begin{array}{ccc}
\lambda((a,1)) = 0 & \lambda((b,1)) = 0 & \lambda((c,1)) = 1 \\
\lambda((a,A)) = 1 & \lambda((b,A)) = 1 & \lambda((c,A)) = 0
\end{array}
$$

$$
\begin{array}{ccc}
\pi((a,1),0) = (c,1) & \pi((b,1),0) = (b,1) & \pi((c,1),0) = (a,1) \\
\pi((a,1),1) = (b,1) & \pi((b,1),1) = (c,1) & \pi((c,1),1) = (a,A) \\
\pi((a,A),0) = (c,A) & \pi((b,A),0) = (b,A) & \pi((c,A),0) = (a,1) \\
\pi((a,A),1) = (b,A) & \pi((b,A),1) = (c,A) & \pi((c,A),1) = (a,A)
\end{array}
$$

In the above example, we have constructed the automaton $\tilde{L}$ in Figure 11a. The automaton appearing in Figure 11b will be explained later. $\triangle$



(a) $L_{A,B}$

(b) The delayed automaton $\sigma L_{A,B}$ is invertible

Figure 11: Construction of $\mathcal{L}_{A,B}$ where A is the adding machine and B is Bellaterra

7 Automatic Exp and Logarithm of Products

It is worthwhile to consider the operation opposite to constructing $\mathrm{Log}_A(B)$.

Let $|X| = 2$, and let $\psi : X^{\mathbb{N}} \to \mathbb{Z}_2$ be the function that naturally identifies words in X with dyadic integers:

$$\psi(w) = \sum_{i=0}^{|w|-1} w_i 2^i.$$

Proposition 7.1 *Let A, B be tree endomorphisms. Define a function* $\mathrm{Exp}_A(B)_n$ *on words w of length n by*

$$Exp_A(B)_n(w) = A^{\psi(B(w))}(w).$$

Then for all n, $\mathrm{Exp}_A(B)_n$ *is an endomorphism of finite trees.*

The endomorphisms of finite trees $\mathrm{Exp}_A(B)_m$, $\mathrm{Exp}_A(B)_k$ *agree on the levels* $1, 2, \ldots, \min(m, k)$ *on which they are both defined.*

Proof. We need to show that if $w \in X^*$ and $x, y \in X$, then $Exp_A(B)(wx)$ and $Exp_A(B)(wy)$ only differ in their last symbol. Let $n = |w|$. Observe that $\psi(B(wx))$ differs from $\psi(B(w))$ by a multiple of 2^n. Now

$$\mathrm{Exp}_A(B)_{n+1}(wx) = A^{a \cdot 2^n} A^{\psi(B(w))}(wx), \tag{4}$$

where $a \in \{0, 1\}$ is given by $a = B|_w(x)$. Note that $A^{2^n}(v) = v$ for any word v of length n because the length of any orbit of A on the level n is a factor of 2^n. Therefore the prefix of length n of $\mathrm{Exp}_A(B)_{n+1}(wx)$ is given by $A^{\psi(B(w))}(w)$, i.e., it does not depend on x. Thus $\mathrm{Exp}_A(B)$ is an endomorphism.

The above argument also shows that $\mathrm{Exp}_A(B)_{n+1}$ and $\mathrm{Exp}_A(B)_n$ agree on levels $1, 2, \ldots, n$. This completes the proof. $\square$

Definition 7.2 *Let* $\mathrm{Exp}_A(B)$ *denote the extension of the maps* $\mathrm{Exp}_A(B)_n$ *to the boundary of the tree $\partial\mathcal{T}$. We shall use the same notation for action on finite words.*

Proposition 7.3 *Let A be a tree endomorphism, and B be a Moore machine. Then* $\mathrm{Exp}_A(B)$ *is an automorphism of the tree $\mathcal{T}$.*

Proof. It suffices to show that $\mathrm{Exp}_A(B)_n$ is invertible for all n. Consider an arbitrary word $w \in X^*$ of length n and a letter $x \in X$. Recall that B being a Moore machine means that $B|_w$ is constant on X. Therefore the value of $\mathrm{Exp}_A(B)_{n+1}(wx)$ (see equation (4)) is given by a power of A that does not depend on x. By assumption, A is invertible, so its sections are invertible as well. Hence $\mathrm{Exp}_A(B)_{n+1}|_w$ acts as a permutation of X.

The proposition follows by induction on n. $\square$

Remark: We have constructed the Log automaton $\mathrm{Log}_A B$ for any invertible Mealy machine B and any level-transitive automaton A of bounded activity. By construction, the Log automaton of an invertible automaton is a Moore machine.

Therefore every invertible automaton B can be written in the form $B = \mathrm{Exp}_A M$, where A is the adding machine (or any bounded-activity, level-transitive automaton), and M is a Moore machine. Note that, in general, one cannot construct a Moore machine (synchronously) equivalent to a given Mealy machine. This construction provides an alternative.

7.1 Logarithm of product

Proposition 7.4 *Let A and B be finite state automata. Then*

$$\mathrm{Exp}_A(\mathrm{Log}_A(B)) = B$$

as endomorphisms of the tree $\mathcal{T}$. In particular, the Automatic Logarithm, as an inverse of Exp, *is unique. That is, if* $\mathrm{Exp}_A B_1 = \mathrm{Exp}_A B_2$, *then $B_1 = B_2$ as endomorphisms of trees.*

Proof. $\mathrm{Exp}_A(\mathrm{Log}_A(B)) = B$ by construction. If $\mathrm{Exp}_A B_1 = \mathrm{Exp}_A B_2$, then for any word w we have $\psi(B_1(w)) = \psi(B_2(w)) \mod 2^{|w|}$. This implies $B_1(w) = B_2(w)$. $\square$

We can now argue about Log using Exp. To proceed, we define:

Definition 7.5 *Let $A = (S_A, \pi_A, \lambda_A, S_{A_0})$ and $B = (S_B, \pi_B, \lambda_B, S_{B_0})$ be finite automata. The **sum automaton** $A \oplus B$ is the automaton with the set of states $S = S_A \times S_B \times \{0, 1\}$, and transition map π and output map λ given by*

$$\begin{aligned} \pi((s,t,c),x) &= (\pi_A(s,x), \pi_B(t,x), d), \ \textit{where} \\ d &= \begin{cases} 1 & \textit{if } \lambda(s,x) + \lambda(t,x) + c \geq 2, \\ 0 & \textit{otherwise}; \end{cases} \\ \lambda((s,t,c),x) &= \lambda(s,x) + \lambda(t,x) + c \mod 2. \end{aligned}$$

For a finite word w, the sum automaton $A \oplus B$ outputs $\psi(A(w)) + \psi(B(w))$ as a dyadic integer. The third component of a state can be understood as the carry bit.

This definition allows us to compute the Log automaton of a product.

Proposition 7.6 *Let B, C be invertible finite automata and A be bounded-activity, level-transitive automaton. Then*

$$\mathrm{Log}_A(BC) = ((\mathrm{Log}_A B)\, C) \oplus \mathrm{Log}_A C.$$

Proof. Let $\mathrm{Log}_A B = a$ and $\mathrm{Log}_A C = c$. Then

$$\begin{aligned} C(w) &= A^{\psi c(w)}(w) \\ BC(w) &= A^{\psi bC(w)}(C(w)) \\ &= A^{\psi bC(w)}(A^{\psi c(w)}(w)) \\ &= A^{\psi bC(w) + \psi c(w)}(w) \\ &= A^{\psi((bC) \oplus c)(w)}(w) \\ &= \mathrm{Exp}_A((bC) \oplus c)(w). \end{aligned}$$

Therefore, by Proposition 7.4,

$$\mathrm{Log}_A(BC) = (bC) \oplus c,$$

which completes the proof. $\square$

8 Distribution of lengths of chords

We now approach the main goal of our investigation. The measure we are interested in is $\mu = \mu_{A,B} := \mathrm{Log}_A(B)_*\nu$, where ν is the uniform Bernoulli measure on $\mathcal{T}$.

This measure gives the distribution of the displacement function: d – a finite integer written in binary as $w = w_0 \dots w_{n-1}$ (and thus interpreted as an element of $\mathbb{Z}/2^n\mathbb{Z}$),

$$\mu(wX^{\mathbb{N}}) = |\{v \in X^n \ : \ \mathrm{Log}_{A,n}(B)(v) = w\}|.$$

We introduce this measure with the goal of studying the properties of the graphs of action, such as their diameter. For example, Pak and Malyshev prove in [**PakMal**] that the diamter of the graph of action of the states of automaton F on level n grows at a rate of $O(n^2)$. However computer experiments give hope that this bound can be improved to $O(n)$. Finding the connections between the measure μ and the properties of the graphs nevertheless remains an open problem.

Figure 3 illustrates the graphs of action with the cycle generated by the adding machine $\mathcal{A}$ put on a circle, and the edges corresponding to the action of another automaton being chords in that circle, motivating the title of this section. The graph on the right has a smaller diameter.

We now proceed to examine interesting properties of μ, and answer questions about it: what kind of measure is μ? Is it Markov, for example?

In fact, there is an easy sufficient condition for μ to be not only Markov, but uniform Bernoulli on a cylinder. To state it, we need to make several definitions:

Definition 8.1 *$\sigma : X^{\mathbb{N}} \to X^{\mathbb{N}}$ is the (left)* ***shift****, defined by $\sigma(aw) = w$ for $a \in X$ and $w \in X^{\mathbb{N}}$. We define the left shift $\sigma : X^* \to X^*$ for finite words w in the same way.*

Definition 8.2 *When $\mathcal{L}$ is a Moore machine, the* ***delayed automaton*** *$\sigma\mathcal{L}$ is the automaton that computes the composition $\sigma \circ \mathcal{L}$. It has the same states, initial state and the transition function as $\mathcal{L}$, but the output function $\sigma\lambda$ is given by*

$$\sigma\lambda(s, x) = \lambda(\pi(s, x)),$$

which is well-defined when $\mathcal{L}$ is a Moore machine.

When $\mathcal{L}$ is Moore, for any finite word $w \in X^*$ and $x \in X$,

$$\mathcal{L}(wx) = \mathcal{L}(0)\sigma\mathcal{L}(w) = \mathcal{L}(1)\sigma\mathcal{L}(w).$$

Proposition 8.3 *Let X be a finite alphabet. Let $\mathcal{L}$ be a Moore machine with initial state s_0, and let $a = \lambda(s_0)$. Let ν be the uniform Bernoulli measure on $X^{\mathbb{N}}$.*

Then $\mu = \mathcal{L}_*\nu$ is supported on the cylinder $aX^{\mathbb{N}}$, and $\mu|_a = (\sigma\mathcal{L})_*\nu$. If σL is invertible, $\mu|_a$ is uniform Bernoulli (i.e. $\mu|_a = \nu$).

Proof. First, note that

$$\mu(aX^{\mathbb{N}}) = \mathcal{L}_*\nu(aX^{\mathbb{N}}) = \nu(\mathcal{L}^{-1}(aX^{\mathbb{N}})) = \nu(\lambda_{s_0}^{-1}(a)X^{\mathbb{N}}) = \nu(X^{\mathbb{N}}) = 1.$$

Now $\mu_a = (\mathcal{L}_*\nu)|_a = (\sigma L)_*\nu$, since for all $v \in X^*$,

$$\begin{aligned}
(\sigma L)_*\nu(vX^{\mathbb{N}}) &= \nu\left((\sigma L)^{-1}(vX^{\mathbb{N}})\right) \\
&= \nu\left(\mathcal{L}^{-1}(\sigma^{-1}\left((vX^{\mathbb{N}})\right)\right) \\
&= \nu\left(\mathcal{L}^{-1}\left(\bigsqcup_{x\in X} xvX^{\mathbb{N}}\right)\right) \\
&= \nu\left(\mathcal{L}^{-1}(avX^{\mathbb{N}})\right) \\
&= \mathcal{L}_*v\left(avX^{\mathbb{N}}\right) \\
&= (\mathcal{L}_*v)|_a(vX^{\mathbb{N}}) \qquad\qquad \text{(since, as noted, } \mathcal{L}_*\nu(aX^{\mathbb{N}}) = 1).
\end{aligned}$$

Thus $(\sigma\mathcal{L})_*\nu = (\mathcal{L}_*\nu)|_a$.

If $\sigma\mathcal{L}$ is invertible, then $(\sigma\mathcal{L})_*\nu = \nu$ by Proposition 4.3. This completes the proof. □

Corollary 8.4 *Let X, A, B and $\mathcal{L} = \mathcal{L}_{A,B}$ be as in Prop 6.3 (so B is invertible, and $\mathcal{L} = \mathcal{L}_{A,B}$ is Moore). Let ν be the uniform Bernoulli measure on X^*.*

Then $\mu = \mathrm{Log}_A(B)_*\nu$ is supported on $\mathcal{L}(0)X^{\mathbb{N}}$, and $\mu|_{\mathcal{L}(0} = \nu$.

Example 8.5 *Let $\mathcal{L} = \mathcal{L}_{A,B}$ with A being the adding machine, and B being the Bellaterra automaton defined in Fig. 10. Then $\mathcal{L}(0) = \mathcal{L}(1) = 0$. The delayed automaton $\sigma\mathcal{L}$ is shown in Figure 11b, and it is invertible (but not minimal: can be reduced to an automaton with 5 states).*

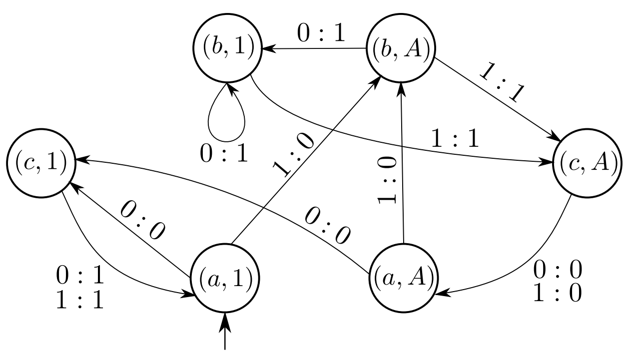
Therefore, $\mu = \mathrm{Log}_A(B)_*\nu$ is the uniform Bernoulli measure supported on $0X^{\mathbb{N}}$, i.e. $\mu|_0 = \nu$ and $\mu|_1 = 0$. △

Proposition 8.3 demonstrates that when B is invertible, the delayed automaton $\sigma\mathcal{L}_{A,B}$ can be useful for examining $\mu_{A,B}$. We make use of it again for what follows:

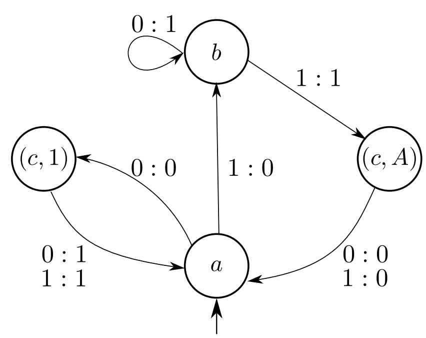
Theorem 8.6 *Let ν be the uniform Bernoulli measure. In the case A is the adding machine and B is automaton F (see Figure 2b), the measure $\mu_{A,B} = \mathrm{Log}_A(B)_*\nu$ is finite-state. Furthermore, $\mu_{A,B}|_0 = 0$, and automaton in Figure 13 computes $\mu_{A,B}|_1$ (in the sense of Definition 3.5).*

Proof. Write $\mu = \mu_{A,B}$. By Proposition 8.3 and the already computed $\mathcal{L} = \mathcal{L}_{A,B}$ in Fig 9, $\mu|_0 = 0$, and the measure is supported on the cylinder $1X^{\mathbb{N}}$, with $\mu|_1 = (\sigma\mathcal{L})_*\nu$. We thus point our attention to $\sigma\mathcal{L}$, shown in Figure 12a.

First, observe that the automaton $\sigma\mathcal{L}$ is not minimal. After identifying states $(a,1)$ and (a,A) into state a, and identifying states $(b,1)$ and (b,A) into state b, we obtain a minimal automaton $\mathcal{L}$ (Figure 12b).



(a) $\sigma\mathcal{L}_{A,B}$



(b) $\sigma\mathcal{L}_{A,B}$ minimized.

Figure 12: Automatons $\sigma\mathcal{L}_{A,B}$ and its minimization

Recall that the states a, b, $(c,1)$ and (c,A) of the automaton $\sigma\mathcal{L}_{A,B}$ are sections of $\mathcal{L}_{A,B}$, and can be seen as endomorphisms whose automata coincide with $\mathcal{L}_{A,B}$ except for the initial state (see Remark 2.7), i.e., $\mathcal{L}_{A,B} = a$, $\mathcal{L}_{A,B}|_1 = b$, etc.

If g is an action on the tree $\mathcal{T}$, we write μ_g for $g_*\nu$. Thus we are interested in $\mu_a = \mu = a_*\nu = (\sigma\mathcal{L})_*\nu$, and we compute it by writing down its sections in terms of μ_a, μ_b, $\mu_{c,1}$ and $\mu_{c,A}$.

We apply Corollary 4.2 to $\mathcal{L}_{A,B}$ to obtain the sections of μ by $x \in X = \{0,1\}$. On the right, we evaluate these measures on the cylindrical sets of the form $xX^{\mathbb{N}}$, so that we could continue the computation by applying Proposition 2.24.

$$
\begin{aligned}
\mu_a|_0 &= \frac{\mu_b + \mu_{c,1}}{2} & \mu_a(0X^{\mathbb{N}}) &= 1 \\
\mu_a|_1 &= 0 & \mu_a(1X^{\mathbb{N}}) &= 0 \\
\mu_b|_0 &= 0 & \mu_b(0X^{\mathbb{N}}) &= 0 \\
\mu_b|_1 &= \frac{\mu_b + \mu_{c,A}}{2} & \mu_b(1X^{\mathbb{N}}) &= 1 \\
\mu_{c,1}|_0 &= 0 & \mu_{c,1}(0X^{\mathbb{N}}) &= 0 \\
\mu_{c,1}|_1 &= \mu_a & \mu_{c,1}(1X^{\mathbb{N}}) &= 1 \\
\mu_{c,A}|_0 &= \mu_a & \mu_{c,A}(0X^{\mathbb{N}}) &= 1 \\
\mu_{c,A}|_1 &= 0 & \mu_{c,A}(1X^{\mathbb{N}}) &= 0
\end{aligned}
$$

Having expressed the sections by one character in terms of each other, we have obtained a set of recursive relations which allows us to compute sections by arbitrary words. To find the set of all sections, we proceed by repeatedly computing sections using Proposition 2.24. We find:

$$
\begin{aligned}
\frac{\mu_b + \mu_{c,1}}{2}|_0 &= 0 & \frac{\mu_b + \mu_{c,1}}{2}(0X^{\mathbb{N}}) &= 0 \\
\frac{\mu_b + \mu_{c,1}}{2}|_1 &= \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4} & \frac{\mu_b + \mu_{c,1}}{2}(1X^{\mathbb{N}}) &= 1 \\
\frac{\mu_b + \mu_{c,A}}{2}|_0 &= \mu_a & \frac{\mu_b + \mu_{c,A}}{2}(0X^{\mathbb{N}}) &= \frac{1}{2} \\
\frac{\mu_b + \mu_{c,A}}{2}|_1 &= \frac{\mu_b + \mu_{c,A}}{2} & \frac{\mu_b + \mu_{c,A}}{2}(1X^{\mathbb{N}}) &= \frac{1}{2}
\end{aligned}
$$

And again:

$$
\begin{aligned}
\frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}|_0 &= \frac{\mu_a + \mu_b + \mu_{c,1}}{3} & \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}(0X^{\mathbb{N}}) &= \frac{3}{4} \\
\frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}|_1 &= \frac{\mu_b + \mu_{c,A}}{2} & \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4}(1X^{\mathbb{N}}) &= \frac{1}{4}
\end{aligned}
$$

Finally:

$$
\begin{aligned}
\frac{\mu_a + \mu_b + \mu_{c,1}}{3}|_0 &= \frac{\mu_b + \mu_{c,1}}{2} & \frac{\mu_a + \mu_b + \mu_{c,1}}{3}(0X^{\mathbb{N}}) &= \frac{2}{3} \\
\frac{\mu_a + \mu_b + \mu_{c,1}}{3}|_1 &= \frac{\mu_b + \mu_{c,A} + 2\mu_a}{4} & \frac{\mu_a + \mu_b + \mu_{c,1}}{3}(1X^{\mathbb{N}}) &= \frac{1}{3}
\end{aligned}
$$

Since we have obtained no new sections at this step, the sections so far are all the sections of μ. We have all the data now to build the automaton in Figure 13 that computes $\mu|_1$. $\square$

The preceding example shows that $\mu_{A,B}$ is finite-state (in the sense of Definition

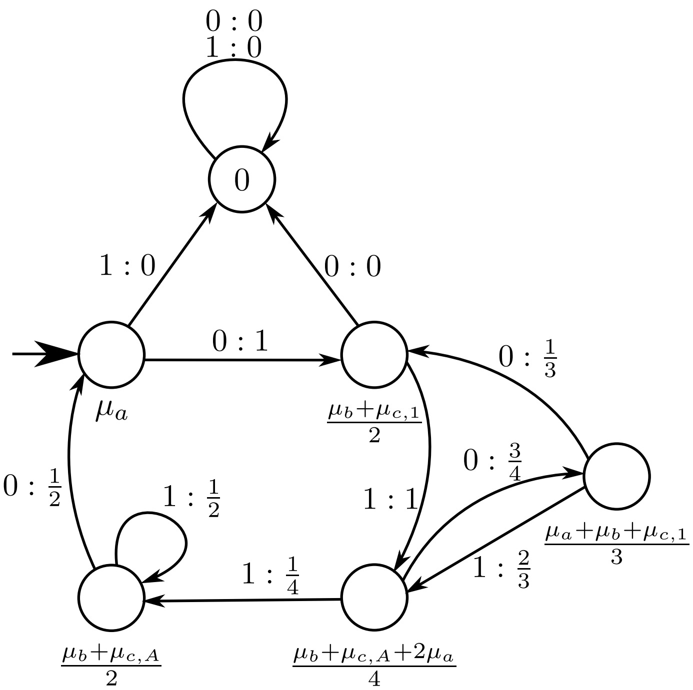


Figure 13: Automaton that computes $\mu_{A,B}|_1$ for A the adding machine and B – automaton F, defined in Figure 2b

3.5) in the case when A is the adding machine and B is automaton F. It should be noted that for some choices of automaton B the measure $\mu_{A,B}$ is not finite-state.

Example 8.7 *Let A be the adding machine and B be the Lamplighter automaton; see Figure 14. Then the measure $\mu_{A,B}$ is not finite-state as shown below.* $\triangle$

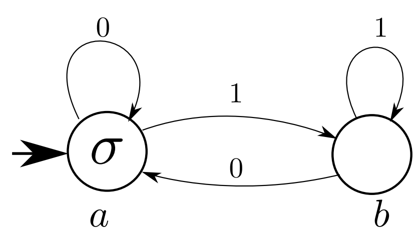
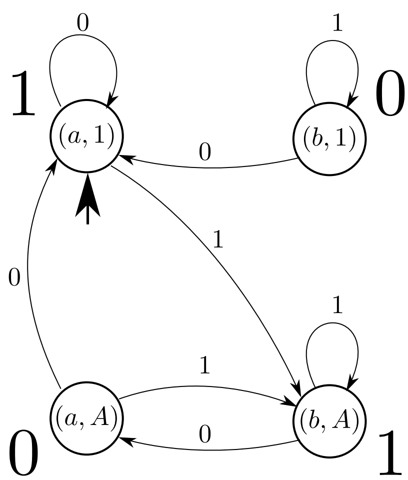


Figure 14: The lamplighter automaton

We compute the automaton $\mathcal{L}_{A,B}$ using Theorem 6.3:

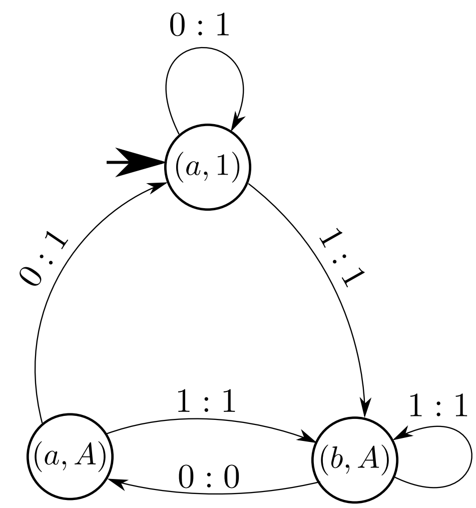
$$\begin{array}{llll} & & \pi((a,1),0) = (a,1) & \pi((b,1),0) = (a,1) \\ \lambda((a,1)) = 1 & \lambda((b,1)) = 0 & \pi((a,1),1) = (b,A) & \pi((b,1),1) = (b,1) \\ \lambda((a,A)) = 0 & \lambda((b,A)) = 1 & \pi((a,A),0) = (a,1) & \pi((b,A),0) = (a,A) \\ & & \pi((a,A),1) = (b,A) & \pi((b,A),1) = (b,A) \end{array}$$

The diagrams of the automata $\mathcal{L}_{A,B}$ and $\sigma\mathcal{L}_{A,B}$ are shown in Figures 15a and 15b, respectively. Since $(b,1)$ is not reachable from the initial state $(a,1)$, it is omitted in Figure 15b. The automaton in that figure is not minimal; states $(a,1)$ and (a,A) can be identified. The minimized automaton is shown in Figure 15c; the relabeling is $a = (a,1) = (a,A)$, $b = (b,A)$, and $(b,1)$ is discarded as unreachable from the initial state a.



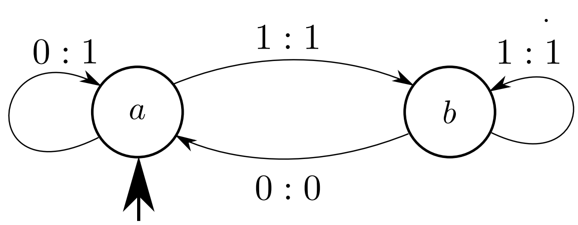
(a) $\mathcal{L}_{A,B}$

.



(b) $\sigma\mathcal{L}$; states unreachable from the initial state $(a,1)$ not shown

.



(c) $\sigma\mathcal{L}$ minimized

.

Figure 15: $\mathcal{L} = \mathcal{L}_{A,B}$ and $\sigma\mathcal{L}$ for A the adding machine, B the Lamplighter.

Noting that $\mu_{A,B}$ is supported on $1X^{\mathbb{N}}$ (by Proposition 8.3), we now point our attention to the measure $\tilde{\mu} = \mu_{A,B}|_1$. Using Corollary 4.2 for the minimized $\sigma\mathcal{L}$ in Figure 15c and the notation of Example 8.6, we get:

$$\begin{aligned} \mu_a|_0 &= 0 & \mu_a(0X^{\mathbb{N}}) &= 0 \\ \mu_a|_1 &= \frac{1}{2}(\mu_a + \mu_b) & \mu_a(1X^{\mathbb{N}}) &= 1 \\ \mu_b|_0 &= \mu_a & \mu_b(0X^{\mathbb{N}}) &= \frac{1}{2} \\ \mu_b|_1 &= \mu_b & \mu_b(1X^{\mathbb{N}}) &= \frac{1}{2} \end{aligned}$$

Now let $\mu_0 := \mu_a$ and $\mu_n := \mu_{n-1}|_1$. Again we use Corollary 2.24:

$$\begin{aligned} \mu_1 &= \frac{(\mu_a + \mu_b)}{2} \\ \mu_2 &= \mu_1|_1 = \frac{1}{2}\left(\mu_a|_1 + \frac{\mu_b|_1}{2}\right)/\mu_1\left(1X^{\mathbb{N}}\right) \\ &= \frac{(\mu_a + 2\mu_b)}{4} \cdot \frac{4}{3} \\ &= \frac{(\mu_a + 2\mu_b)}{3} \\ \mu_3 &= \mu_2|_1 = \frac{(\mu_a + 3\mu_b)}{4} \\ &\dots \\ \mu_n &= \mu_{n-1}|_1 = \dots \end{aligned}$$

All this leads to the following:

Proposition 8.8 *Let $\mu_0 = \mu_a$, and $\mu_n = \mu_{n-1}|_1$, for $n \in \mathbb{N}$. Then*

$$\begin{aligned} \mu_n &= \frac{\mu_a + n\mu_b}{n+1} \\ \mu_n(0X^{\mathbb{N}}) &= \frac{n}{2(n+1)} \\ \mu_n(1X^{\mathbb{N}}) &= \frac{n+2}{2(n+1)} \end{aligned}$$

Proof. By induction on n. The proposition holds for $n = 0$. Assuming it holds for

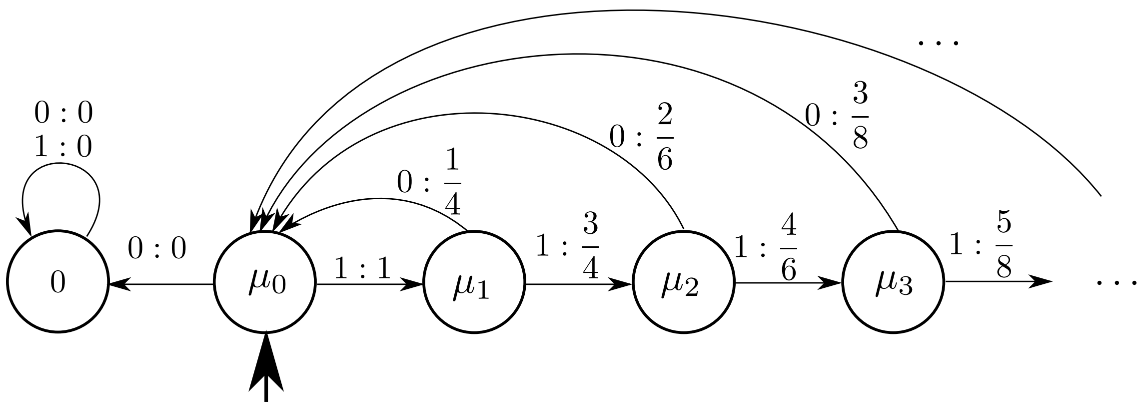


Figure 16: The infinite automaton computing $\tilde{\mu}_{A,B}$ where A is the adding machine, and B is the Lamplighter automaton.

$n = k$, by Corollary 2.24:

$$\begin{aligned} \mu_{k+1} = \mu_k|_1 &= \frac{1}{k+1}\left(\frac{\mu_a + \mu_b}{2} + \frac{1}{2}k\mu_b\right) / \left(\frac{k+2}{2(k+1)}\right) \\ &= \frac{\mu_a + (k+1)\mu_b}{k+2}. \ \square \end{aligned}$$

Note that measures μ_n are all distinct.

Corollary 8.9 *$\mu_{A,B}$ is not finite-state when A is the adding machine and B is Lamplighter.*

Corollary 8.10 *μ_n for $n = 0, 1, 2, \ldots$ are all the nontrivial sections of $\tilde{\mu}$.*

Proof. This immediately follows from observing that $\mu_n|_0 = \mu_0$ for $n > 0$:

$$\mu_n|_0 = \frac{\mu_a + n\mu_b}{n+1}|_0 = \frac{1}{n+1}\frac{n\mu_a}{2}\frac{2(n+1)}{n} = \mu_a = \mu_0.$$

The (infinite) automaton that computes $\tilde{\mu}$ is shown in Figure 16. $\square$

Observe that the computations in these examples are almost linear. The following proposition makes this notion precise.

Proposition 8.11 *Let $X = \{x_0, \dots, x_{k-1}\}$ be a finite alphabet, L be a Mealy machine with states $S = \{g_0, \dots, g_{n-1}\}$, and ν be a Bernoulli measure given by a vector $p = (p(x_0), \dots, p(x_{k-1}))$. For any vector $v = (a_0, a_1, \dots, a_{n-1}) \in \mathbb{R}$ let*

$$\mu_v = \sum_{i=0}^{n-1} a_i g_{i*}\nu.$$

Then for any $x \in X$ there exists an $n \times n$ matrix M_x and an n-dimensional vector p_x such that

$$\mu_v|_x = \mu_w$$

with

$$w = \frac{M_x v}{p_x . v}.$$

The entries of M_x and coordinates of p_x are given by

$$M_x(i,j) = \sum_{y:\pi(g_i,y)=g_j \text{ and } \lambda(g_i,y)=x} p(y);$$

$$p_x(j) = \sum_{i=0}^{n-1} M_x(i,j).$$

Proof. From Proposition 4.2 and Corollary 2.24:

$$\left.\left(\sum_{i=0}^{n} a_i g_{i*}\nu\right)\right|_x = \frac{\sum_{i=0}^{n} a_i g_{i*}\nu(xX^{\mathbb{N}})(g_{i*}\nu)|_x}{\sum_{i=0}^{n} a_i g_{i*}\nu(xX^{\mathbb{N}})}$$

$$= \frac{\sum_{i=0}^{n} a_i \sum_{y \in \lambda_{g_i}^{-1}(x)} p(y)\pi(g_i,y)_*\nu}{\sum_{i=0}^{n} a_i \sum_{y \in \lambda_{g_i}^{-1}(x)} p(y)}.$$

The proposition follows. $\square$

Corollary 8.12 *Let*

$$\phi_x(v) := \frac{M_x v}{p_x \cdot v}.$$

Then μ_v is finite-state if and only if the orbit of v under the action of the semigroup generated by ϕ_x, $x \in X$ is finite. The graph of the action is the transition diagram of the automaton that computes $\mu_{[v]}$.

The above corollary can be made simpler once we consider v as an element of $\mathbb{RP}^n$. For $v = (a_0, a_1, \dots, a_{n-1})$, write $[v] = [a_0 : a_1 : \dots : a_{n-1}] \in \mathbb{RP}^n$ and let

$$\mu_{[v]} := \frac{\mu_v}{\mu_v\left(X^{\mathbb{N}}\right)}.$$

This is well defined and

$$[\phi_x(v)] = [M_x v].$$

Corollary 8.13 *$\mu_{[v]}$ is finite-state if and only if the orbit of $[v]$ under the action of the semigroup generated by $\langle M_x : x \in X\rangle$ is finite.*

In the special case when ν is the uniform Bernoulli measure, it is convenient to use matrices $\tilde{M}_x$ with entries

$$\tilde{M}_x(i,j) = \sum_{y:\pi(g_i,y)=g_j \text{ and } \lambda(g_i,y)=x} 1.$$

Similarly, set $\tilde{p}_x = |X| p_x$. By definition, $\tilde{M}_x = |X| M_x$, $[\tilde{M}_x v] = [M_x v]$, and $\phi_x(v) = \tilde{M}_x v / \tilde{p}_x \cdot v$. However $\tilde{M}_x$ has integer entries: $M_x(i,j) \in \{0, 1, \dots, |X|\}$.

Corollary 8.14 *In the case ν is uniform Bernoulli, the measure $\mu_{[v]}$ is finite-state if and only if the orbit of $[v]$ under the action of the multiplicative semigroup generated by integer matrices $\tilde{M}_x$, $x \in X$ is finite.*

Example 8.15 *When $L = L_{A,B}$, where A is the adding machine and B is automaton F given by Figure 2b, we have*

$$\begin{aligned}
\tilde{M}_0 &= \begin{pmatrix} 0 & 0 & 0 & 2 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \\
\tilde{p}_0 &= (2, 0, 0, 2) \\
\phi_0(v) &= \tilde{M}_0 v / \tilde{p}_0 \cdot v \\
\tilde{M}_1 &= \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \\
p_1 &= (0, 2, 2, 0) \\
\phi_1(v) &= M_1 v / p_1 \cdot v
\end{aligned}$$

The orbit of $(1,0,0,0)$ *under the action of* $\langle \phi_0, \phi_1 \rangle$ *is*

$$((0,0,0,0),(0,1/2,0,1/2),(0,1/2,1/2,0),(1/3,1/3,1/3,0),(1/2,1/4,0,1/4),(1,0,0,0)).$$

These correspond to the states in Figure 13.

Equivalently, the orbit of $[1:0:0:0]$ *under the action of* $\langle \tilde{M}_0, \tilde{M}_1 \rangle$ *is*

$$([0:0:0:0],[0:1:0:1],[0:1:1:0],[1:1:1:0],[2:1:0:1],[1:0:0:0]).$$

$\triangle$

Example 8.16 *When* $L = L_{A,B}$ *with* A *the adding machine and* B *the Lamplighter, we have*

$$\tilde{M}_0 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \qquad \tilde{M}_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

The orbit of $[1:0]$ *under the action of* $\tilde{M}_1$ *is* $\{[1:n] \ : \ n \in \mathbb{N}\}$*, and is not finite.*
$\triangle$

9 When the measure is Markov

Having obtained the automaton that computes a measure, we can ask the question of what kind of measure it is. Recall from Definition 3.3 that a k-step Markov measure is a measure whose sections are uniquely determined by *suffixes* of length k, regardless of what comes before. The following theorem provides necessary and sufficient conditions for a finite-state measure to be k-step Markov.

Theorem 9.1 *Let* μ *be a finite-state measure with* n *nontrivial sections* $\mu_1, \dots, \mu_n$*. Then* μ *is* k*-step Markov (for some* $k \in \mathbb{N}$*) if and only if for any nonempty word* $w \in X^*$*, there is at most one* i*,* $1 \leq i \leq n$ *such that* $\mu_i|_w = \mu_i$*. When* μ *is* k*-step Markov, we can choose* $k \leq n(n-1)$*.*

Proof. $\Rightarrow$ Proof by contradiction. Let μ be k-step Markov. Assume that the hypothesis of the theorem does not hold, that is, μ has two distinct nontrivial sections $\mu|_u$ and $\mu|_v$ (where $u, v \in X^*$) such that $(\mu|_u)|_w = \mu|_u$ and $(\mu|_v)|_w = \mu|_v$ for some nonempty word $w \in X^*$. Let $W = www\dots w$ be the word w repeated several times so that $|W| > k$. Then

$$\mu|_{uW} = (\mu|_u)|_W = (\mu|_u)|_{ww\dots w} = \mu|_u,$$
$$\mu|_{vW} = (\mu|_v)|_W = (\mu|_v)|_{ww\dots w} = \mu|_v.$$

So the nontrivial sections $\mu|_{uW}$ and $\mu|_{vW}$ are different, but $|W| > k$. That is, a suffix of length k does not uniquely determine a nontrivial section of μ. This contradicts the assumption that μ is k-Markov.

$\Leftarrow$: Assume now that the hypothesis holds. For μ to be k-step Markov, it suffices to show that for any two nontrivial sections $\mu|_u$ and $\mu|_v$, and any word w with $|w| = k$, we have $\mu|_{uw} = \mu|_{vw}$ whenever both uw and vw are admissible words.

Let $k = n(n-1)$ and fix $w = w_1 w_2 \dots w_k$, $w_i \in X$ with $|w| = k$. Consider Table 3. The columns of this table are the paths from $\mu|_u$ and $\mu|_v$ obtained by taking sections by the word w character by character. By assumption, uw and vw are admissible,

$\mu\|_u$	$\mu\|_v$
$\mu\|_{uw_1}$	$\mu\|_{vw_1}$
$\mu\|_{uw_1w_2}$	$\mu\|_{vw_1w_2}$
$\cdots$	$\cdots$
$\mu\|_{uw}$	$\mu\|_{vw}$

Table 3: Paths of length k starting from $\mu|_u$ and $\mu|_v$

so Table 3 contains only nontrivial sections. We claim that one row of the table contains two identical measures: $\mu|_{uw_1..w_i} = \mu|_{vw_1..w_i}$ for some i, $0 \le i \le k$. Then each subsequent row also contains two identical measures. In particular, $\mu|_{uw} = \mu|_{vw}$. To prove the claim (and complete the proof), assume the contrary. Since μ has only n nontrivial sections, there are only $n(n-1)$ pairs of distinct nontrivial sections. Table 3 has $k+1 > n(n-1)$ rows, hence a row in the table must repeat, i.e.,

$$(\mu|_{uw_1..w_i}, \mu|_{vw_1..w_i}) = (\mu|_{uw_1..w_j}, \mu|_{vw_1..w_j})$$

for some $1 \le i < j \le k$. But that means that the word $W := w_{i+1} \dots w_j$ fixes two sections $\mu_U := \mu|_{uw_0..w_i}$ and $\mu_V := \mu|_{vw_0..w_i}$; that is, $(\mu|_U)|_W = \mu|_U$ and $(\mu|_V)_W = \mu|_V$. By our hypothesis, the nontrivial section fixed by W is unique, so $\mu|_U = \mu_V$. Since $UW = uw$ and $VW = vw$, this implies $\mu|_{uw} = \mu|_{vw}$. This completes the proof. $\square$

Remark: The free semigroup $FS(X)$ generated by X acts on the sections of μ: for $w \in FS(X)$, $w \cdot \mu_i := \mu_i|_w$. The condition of Theorem 9.1 can be re-stated as follows: the action of any nonempty word $w \in X^*$ on the sections of μ has at most one nontrivial fixed point.

Theorem 9.1 is illustrated by the following example.

Example 9.2 *When A is the adding machine and B is automaton F, the measure $\mu_{A,B}$ is defined by the automaton M in Figure 13. The measure satisfies the hypothesis of Theorem 9.1. By the theorem, $\mu_{A,B}$ is k-step Markov for some $k \le 20$. Direct* *examination of the automaton M reveals that the admissible words for $\mu_{A,B}$ are all words not containing* 000 *or* 1101*, and the measure is, in fact,* 3*-step Markov (see Table 4).* $\triangle$

w ends in	$\mu\vert_w$
00	$\frac{\mu\vert_b + \mu\vert_{c,1}}{2}$
11	$\frac{\mu\vert_b + \mu\vert_{c,A}}{2}$
01	$\frac{\mu\vert_b + \mu\vert_{c,A} + 2\mu_a}{4}$
110	μ_a
010	$\frac{\mu_a + \mu\vert_b + \mu\vert_{c,1}}{3}$

Minimal forbidden words:
000, 1101.

Table 4: $\mu_{A,B}$ as a Markov measure when B is automaton F

The condition of Theorem 9.1 is not satisfied trivially.

Example 9.3 *A finite-state measure μ defined by the automaton in Figure 17 is not a Markov measure. The initial state μ (on the top left) is fixed by the action of the word* 01*, but so is the top right state, $\mu|_{001}$: $\mu|_{01} = \mu$ and $(\mu|_{001})|_{01} = \mu|_{001}$. Note that $\mu \neq \mu|_{001}$ since $\mu(0X^{\mathbb{N}}) = \frac{3}{7}$ while $\mu|_{001}(0X^{\mathbb{N}}) = \frac{2}{5}$. By Theorem 9.1, μ cannot be a k-step Markov measure for any k.* $\triangle$

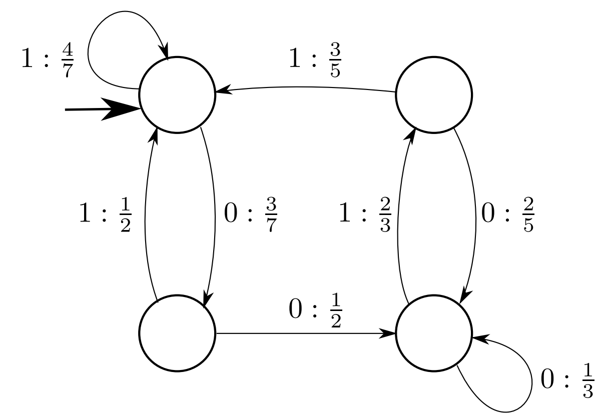


Figure 17: A diagram of the automaton of a finite-state measure that is not Markov.